



UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
UNIDAD DE ESTUDIOS A DISTANCIA
MODALIDAD SEMIPRESENCIAL
CARRERA SISTEMAS

TESIS DE GRADO

**Sistema informático para la gestión y el control de servicios
generales de la Universidad Técnica Estatal de Quevedo.**

AUTOR

JORGE PATRICIO BRAULIO

DIRECTOR

ING. CARLOS CAIZAGUANO

QUEVEDO - LOS RÍOS - ECUADOR

2013

**UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
UNIDAD DE ESTUDIOS A DISTANCIA
CARRERA SISTEMAS**

**SISTEMA INFORMÁTICO PARA LA GESTIÓN Y EL CONTROL DE
SERVICIOS GENERALES DE LA UNIVERSIDAD TÉCNICA ESTATAL DE
QUEVEDO**

Tesis de grado presentada al Comité Técnico Académico Administrativo de la
Unidad de Estudios a Distancia como requisito previo para la obtención del
título de:

INGENIERO EN SISTEMAS

MIEMBROS DEL TRIBUNAL

Ing. Msc. Nancy Rodríguez Gavilanes
PRESIDENTE DEL TRIBUNAL

Ab. Msc. Miriam Cárdenas Zea
MIEMBRO DEL TRIBUNAL

Ing. Msc. Jefferson Bravo Salvatierra
MIEMBRO DEL TRIBUNAL

Ing. Msc. Carlos Caizaguano
DIRECTOR DE TESIS

DECLARACIÓN

Yo, **Jorge Patricio Braulio**, declaro bajo juramento que el trabajo aquí descrito es de mí autoría; que no ha sido previamente presentado 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 el derecho de propiedad intelectual correspondiente a este trabajo a la Unidad de Estudios a Distancia de la Universidad Técnica Estatal de Quevedo, según lo establecido por la ley de propiedad intelectual por su reglamento y por la normatividad institucional vigente.

.....
Jorge Patricio Braulio

AUTOR

CERTIFICACIÓN

Ing. **Carlos Caizaguano**, Docente de la Unidad de Estudios a Distancia de la Universidad Técnica Estatal de Quevedo

Certifica:

Que el egresado **Jorge Patricio Braulio**, realizó la investigación propuesta y ha cumplido a cabalidad con los requisitos de desarrollo de software y el correspondiente informe técnico de la Tesis de Grado titulada “**SISTEMA INFORMÁTICO PARA LA GESTIÓN Y EL CONTROL DE SERVICIOS GENERALES DE LA UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO**”, bajo mi dirección, acatando las disposiciones reglamentarias establecidas para el efecto.

Ing. Carlos Caizaguano Ch.
DIRECTOR DE TESIS

DEDICATORIA

Mis sinceros Agradecimientos al todo poderoso por darme la vida la sabiduría y la paciencia para poder realizar mi tesis.

Jorge Patricio Braulio

AGRADECIMIENTO

A mí querida esposa quien ha sabido comprender todas las horas que pasé fuera de casa.

Agradezco a todas y cada una de las personas que supieron guiarme con el desarrollo de este trabajo de tesis.

Hago extensivo mi sincero agradecimiento de manera especial al Ing. Carlos Caizaguano Director de Tesis quien ha sabido asesorar y compartir sus vastos conocimientos., Ing. Msc. Nancy Rodríguez Gavilanes, a Ing. Msc. Lucrecia Llerena Guevara por el asesoramiento académico brindado.

Jorge Patricio Braulio

Autor.

Índice

CAPÍTULO I.....	10
I. INTRODUCCIÓN	11
1.1. Objetivos	15
1.1.1. General.....	15
1.1.2. Específicos.....	15
1.2. Hipótesis	16
CAPÍTULO II.....	17
2.1. Sistema de Información	18
2.1.1. Que son los Sistemas de Información.....	18
2.1.2. Entrada de Información.....	18
2.1.3. Almacenamiento de información	19
2.1.4. Procesamiento de Información	19
2.1.5. Salida de Información	19
2.1.6. Tipos y Usos de los Sistemas de Información.....	20
2.1.7. Sistemas Transaccionales.....	20
2.1.8. Sistemas de Apoyo de las Decisiones	21
2.1.9. Sistemas Estratégicos	22
2.1.10. Evolución de los Sistemas de Información.....	23
2.1.11. Etapa de contagio o expansión	24
2.1.12. Etapa de control o formalización	25
2.1.13. Etapa de integración.....	26
2.1.14. Etapa de administración de datos	27
2.1.15. Etapa de madurez.....	28
2.2. Administración	28
2.2.1. ¿Qué es un inventario?	30
2.3. Base de Datos	31
2.3.1. Que son las bases de datos.	31
2.3.2. Porque la creación de las bases de datos.	32

2.3.3.	Mejora en la integridad de datos.....	32
2.3.4.	Mejora en la seguridad.....	32
2.3.5.	Abstracción de datos.....	32
2.3.6.	Aumento de la concurrencia.....	33
2.3.7.	Características.....	33
2.3.8.	Vulnerable a los fallos.....	33
2.3.9.	Tipos de Campos.....	34
2.3.10.	Tipos de Base de Datos.....	34
2.3.11.	Bases de datos estáticas.....	35
2.3.12.	Bases de datos dinámicas.....	35
2.3.13.	Bases de datos bibliográficas.....	35
2.3.14.	Bases de datos de texto completo.....	36
2.3.15.	Bases de Datos o "bibliotecas" de información química o biológica.....	36
2.3.16.	Tipos de modelos de Datos.....	36
2.4.	SQL SERVER.....	38
2.4.1.	Introducción a las Bases de Datos de SQL SERVER.....	38
2.4.2.	Almacenamiento de los Datos.....	38
2.4.3.	Archivos de Bases de Datos de SQL Server.....	39
2.4.4.	Tipos de archivos de las Bases de Datos.....	39
2.4.5.	Componentes del SQL.....	39
2.4.5.1.	Comandos.....	40
2.4.5.2.	Cláusulas.....	41
2.4.5.3.	Operadores Lógicos.....	41
2.4.5.4.	Operadores de Comparación.....	42
2.4.5.5.	Funciones de Agregado.....	42
2.5.	Visual Studio .Net.....	43
2.5.1.	Introducción a Visual Studio .NET.....	43
2.5.2.	Microsoft .Net Framework.....	44
2.5.3.	La Plataforma Microsoft .Net.....	44
2.5.4.	Marco .Net.....	45
2.5.5.	Componentes del Marco .Net.....	45
2.5.6.	Common Language Runtime (CLR).....	45
2.5.7.	Tipos de Ejecutables.....	46

2.5.8.	¿Qué es un Namespace (o espacio de nombres)?	46
2.5.9.	¿Qué es un Assembly (o ensamblado)?.....	47
2.5.10.	Librerías de Clases del Marco .Net	47
2.5.11.	Creación de Componentes .NET	48
2.5.12.	Creación de Archivos XML.....	48
2.5.13.	Lenguajes en .NET.....	48
2.6.	Ingeniería de Software	48
2.6.1.	¿Qué es la Ingeniería de Software?.....	48
2.6.2.	Definición de Ingeniería de Software	49
2.6.3.	Calidad del Software	49
2.6.4.	Definición: Ingeniería	50
2.6.5.	Definición de Software.....	50
2.6.6.	Análisis de Sistemas.....	51
2.6.7.	Diseño de Sistemas.	53
2.6.8.	Introducción	54
2.6.9.	Enfoques de desarrollo de software	55
2.6.10.	Metodología de desarrollo OPEN UP.....	55
2.6.11.	Principios de OPEN UP	56
2.6.11.1.	Promover la colaboración para alinear los intereses comunes y difundir el conocimiento.....	57
2.6.11.2.	Balanceo de las prioridades competitivas para maximizar el valor de los participantes del proyecto.	57
2.6.11.3.	Desarrollo de una arquitectura al principio con el fin de minimizar riesgos y planificar el desarrollo.....	57
2.6.11.4.	Evolucionar para obtener un feedback continuo en aras de una mejora continua.....	57
2.6.11.5.	Ciclo de vida de OpenUP	59
2.6.11.6.	Prácticas de OpenUP	60
2.6.11.7.	Roles de OpenUP.....	61
2.6.11.8.	Otros enfoques de desarrollo de software.....	62
2.6.11.9.	Lenguaje Unificado Modelado (UML).....	62
2.6.11.10.	Diagrama de Clases	63
2.6.11.11.	Diagrama de Caso de Uso	63
2.6.11.12.	Diagrama de Secuencia.....	64

2.6.11.13.	Diagrama de Colaboración.....	64
2.6.11.14.	Diagrama de Actividades.....	65
2.6.11.15.	Modelo Conceptual.....	65
2.6.11.16.	Metodología Xtreme Programming	65
2.6.11.17.	Metodología OMT	75
2.6.11.18.	Introducción.	75
2.6.11.19.	Modelo de Objetos.....	77
2.6.11.20.	Clases y Objetos	78
2.6.11.21.	Enlaces y Asociaciones	79
2.6.11.22.	Conceptos Avanzados de Enlaces y Asociaciones.....	80
2.6.11.23.	Generalización y Herencia	81
2.6.11.24.	Agrupación de entidades	83
2.6.11.25.	Modelo Avanzado de Datos	83
2.6.11.26.	Construcción de un modelo de objetos	85
2.6.11.27.	Modelo Dinámico	85
2.6.11.28.	Sucesos y Estados	86
2.6.11.29.	Diagrama de Estados.....	87
2.6.11.30.	Condiciones	89
2.6.11.31.	Operaciones.....	89
2.6.11.32.	Diagramas de Estados Anidados	90
2.6.11.33.	Concurrencia	91
2.6.11.34.	Desarrollo de un modelo dinámico.....	92
2.6.11.35.	Modelo Funcional.....	92
2.6.11.36.	Diagramas de flujo de datos.....	92
2.6.11.37.	Especificación de Operaciones	94
2.6.11.38.	Restricciones	96
2.6.11.39.	Construcción de un modelo funcional.....	97
2.6.11.40.	Fase de Análisis.....	97
2.6.11.41.	Fase de Diseño de sistemas.....	98
2.6.11.42.	Fase de Diseño de objetos.....	99
2.7.	Servicios Generales en la Empresa Moderna	100
2.7.1.	Introducción	100
2.7.2.	La organización de Servicios Generales	102

2.7.3.	Definición de las actividades de SG	103
CAPÍTULO III.....	104	
3.1.	Materiales y Equipos.....	105
3.1.1.	Método Analítico.....	105
3.1.2.	Método Sintético.....	105
3.1.3.	Materiales	105
3.2.	Tipo de Investigación.....	106
3.2.1.	Proyectiva.....	106
3.2.2.	Investigación exploratoria	106
3.2.3.	Investigación Descriptiva.....	107
3.2.4.	Bibliográfica.....	107
3.3.	Fuentes	107
3.3.1.	Primarias	107
3.3.2.	Secundarias.....	107
3.4.	Técnicas e Instrumentos de Evaluación.....	108
3.4.1.	Entrevista	108
3.4.2.	Encuesta.....	108
3.4.3.	Técnicas e Instrucciones de Evaluaciones.....	108
3.5.	Procedimiento Metodológico.....	109
3.5.1.	Concepción.....	109
3.5.2.	Elaboración.....	109
3.5.3.	Construcción.....	109
3.5.4.	Transición	109
CAPÍTULO IV	111	
RESULTADOS Y DISCUSIÓN	111	
4.1.	Situación actual de la Unidad de Servicios Generales.....	112
4.1.1.	Requisitos.....	113
4.1.2.	Análisis.....	113
4.1.3.	Diseño Preliminar.....	113
4.1.4.	Diseño Detallado.....	113
4.1.5.	Implementación y Pruebas.....	113

4.2.	Análisis e Interpretación de Resultados	114
4.2.1.	Resultados de las encuesta a los usuarios que reciben los servicios de la Unidad de Servicios Generales de la UTEQ.....	114
4.2.1.1.	Calificación de la atención a los usuarios	114
Cuadro Nº 1 Calificación de la atención a los usuarios.....		115
4.2.1.2.	Trabajo de manera técnico y profesional.	115
4.2.1.3.	Trabajo de manera eficaz, eficiente y oportuna.	115
4.2.1.4.	Está de acuerdo se automática la USG.	116
4.2.2.	Resultados de las encuesta a los empleados que laboran en la Unidad de Servicios Generales de la UTEQ.....	117
4.2.2.1.	Calificación de la atención que brindan	117
Cuadro Nº 5 Calificación de la atención que brindan		117
4.2.2.2.	Problemas al realizar pedido de materiales.....	117
Cuadro 6. Problemas al realizar pedido de materiales.		118
4.2.2.3.	Informe de daños al asignar trabajo.....	118
4.2.2.4.	¿Está de acuerdo se automatiche la USG?	118
4.2.3.	Resultados de las Entrevistas	119
4.2.3.1.	Entrevista al Coordinador de la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo.	119
4.3.	Discusión	121
4.4.	Propuesta	122
4.4.1.	Concepción	122
4.4.2.	Elaboración	122
4.4.3.	Construcción	123
4.4.4.	Transición	123
4.5.	Documentación del Sistema.....	124
4.5.1.	Casos de Uso	125
4.5.1.1.	Caso de Uso General	125
4.5.1.2.	Casos de Uso Expandido	126
4.5.1.2.1.	Inicio de Sesión	126
4.5.1.2.2.	Crear Cuenta de Usuario	126
4.5.1.2.3.	Registra Personal de Servicio.....	127

4.5.1.2.4. Ingreso de Actividades del Personal.....	127
4.5.1.2.5. Ingreso de Herramienta	128
4.5.1.2.6. Ingreso de Material	128
4.5.1.2.7. Registra Unidad de Medida	129
4.5.1.2.8. Registra Ingreso de Kardex.....	129
4.5.1.2.9. Realizar Consulta.....	130
4.5.1.2.10. Informe de Actividades.....	130
4.5.1.2.11. Informe de Material	131
4.5.1.2.12. Salida de Material	131
4.5.1.2.13. Salida de Herramientas.....	132
4.5.2. Descripción del Caso de Uso Expandido	133
4.5.2.1. Inicio de Sesión	133
4.5.2.2. Configurar Cuenta de Usuario	134
4.5.2.3. Registra Personal de Servicio	135
4.5.2.4. Ingreso de Actividad	136
4.5.2.5. Registra Tipo de Herramienta	137
4.5.2.6. Registra Tipo de Material.....	138
4.5.2.7. Registra Unidad de Medida	139
4.5.2.8. Registra Ingreso de Kardex.....	140
4.5.2.9. Realiza Consulta	141
4.5.2.10. Realiza Informe de Actividad.....	142
4.5.2.11. Realiza Informe de Material.....	143
4.5.2.12. Registra Salida de Material	144
4.5.2.13. Registra Salida de Herramienta.....	145
4.5.2.14. Inicio de Sesión	146
4.5.2.15. Configurar Cuenta de Usuario	146
4.5.2.16. Registra Personal de Servicio.....	147
4.5.2.17. Ingreso de Actividades	147
4.5.2.18. Registra Ingreso de Herramientas	148
4.5.2.19. Registra Ingreso de Material	148
4.5.2.20. Registra Unidad de Medida	149
4.5.2.21. Ingreso de Kardex.....	149
4.5.2.22. Realizar Consulta.....	150

4.5.2.23. Realiza Informe de Actividad.....	150
4.5.2.24. Informe de Material	151
4.5.2.25. Salida de Material	151
4.5.2.26. Realiza Salida de Herramientas	152
4.5.3. Diagrama de Secuencia	153
4.5.3.1. Inicio de Sesión	153
4.5.3.2. Configurar Cuenta de Usuario	153
4.5.3.3. Registra Personal de Servicio	154
4.5.3.4. Ingreso de Actividades	154
4.5.3.5. Registra Ingreso de Herramientas.....	155
4.5.3.6. Registra Ingreso de Material	155
4.5.3.7. Registra Unidad de Medida	156
4.5.3.8. Registra Ingreso de Kardex.....	156
4.5.3.9. Realiza Consulta	157
4.5.3.10. Informe de Actividad	157
4.5.3.11. Informe de Material	158
4.5.3.12. Salida de Material	158
4.5.3.13. Salida de Herramientas.....	159
4.5.4. Diagrama Físico de la Base de Datos	160
4.5.5. Diagrama de Clase	161
CAPÍTULO V	162
5.1. CONCLUSIONES	163
5.2. RECOMENDACIONES	165
BIBLIOGRAFÍA	166
6.1. BIBLIOGRAFÍA.....	167
6.2. Páginas Web.....	167
http://www.gestiopolis.com/dirgp/adm/produccion.htm	169
ANEXOS.....	170
7.1. Cuestionario de Encuestas y Entrevistas	171

7.2. Manual de Usuario	176
7.2.1. Introducción	176
7.2.2. Objetivos del Manual.....	176
7.2.3. Especificaciones Técnicas	176
7.2.4. Funcionamiento del Sistema	177

CAPÍTULO I

MARCO CONTEXTUAL DE LA INVESTIGACIÓN

I. INTRODUCCIÓN

La Universidad Técnica Estatal de Quevedo desde su creación ha mejorado tanto en la preparación de docentes como en la infraestructura de las áreas docente, administrativa y de investigación; equipados con tecnologías de punta, para lograr un reconocimiento por sus resultados en la sociedad fluminense.

La Institución cuenta con algunas infraestructuras entre ellas están los Edificios de: La Facultad de Agrarias, la Facultad de Ambientales, la Facultad de Ciencias Empresariales, el Instituto de Informática, Posgrado, áreas Administrativas, Auditorium, D.B.U., actualmente se encuentra en constante avance con nuevas edificaciones para el desarrollo institucional, que permitan brindar una mejor atención al público en general.

El área de la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo tiene la finalidad de prestar sus servicios de mantenimiento en electricidad, plomería, gasfitería, jardinería, pintura, albañilería, bienes inmuebles, que requieran reparación en todas sus instalaciones.

La Unidad de Servicios Generales realiza mantenimiento a todas las infraestructuras de la UTEQ, previniendo el buen funcionamiento de los inmuebles de manera que puedan perdurar por más tiempo.

En los aires acondicionados la lista de tareas a realizar es más amplio por tal motivo su correcto mantenimiento preventivo es parte fundamental a la hora de generar un plan de trabajo y tener claros los puntos a resolver. El área eléctrica en los departamentos universitarios es uno de los servicios básicos más importantes, requiere de mucha atención, al igual que otro servicio que da mantenimiento en lámparas, interruptores, tomacorrientes, cometidas eléctricos etc.

La plomería es la actividad relacionada con la instalación de redes de abastecimiento de agua potable y evacuación de aguas residuales, así como las instalaciones de calefacción en edificaciones y otras construcciones dentro de una Institución; en mampostería se realizan reparaciones o arreglos de paredes, enlucidos, o en ciertos casos se realizan pequeños trabajos que requieren las instalaciones de la Empresa.

La pintura la cual le da el toque perfecto al acabado de una construcción, es otro servicio que se da mantenimiento. Y la jardinería otro factor importante, el mantenimiento es constante ya que en épocas de verano y en épocas de invierno tienen sus pequeños estragos, dando un mal aspecto al ornato de la Institución.

Debido a que la tecnología está en constante cambio es necesario ir actualizando las infraestructuras, normalmente la evolución tecnológica se ha desarrollado desde los formatos en papel, luego con el advenimiento de las hojas electrónicas se cambiaron las formas del papel a las hojas de cálculo como Excel y posteriormente aparece el uso de aplicaciones con base de datos.

Razón por la cual, el sistema que se pretende desarrollar en esta tesis de grado para el área de Servicios Generales, tiene como objetivo liberar a esta dependencia de los procesos manuales relacionados con el control de mantenimiento de todas las infraestructuras de la UTEQ debido a que este control se manipulan en hojas de Excel y tarjetas kardex sin un sistema que permita satisfacer las demandas actuales de información.

Es de fundamental importancia llevar un registro de los servicios generales solicitado por los jefes departamentales, debido a que el monitoreo permitirá estimar la situación actual de los mismos y tomar correctivos necesarios; pero este seguimiento al realizarlo manualmente, ha ocasionado retrasos en el cumplimiento de lo solicitado, demora de trabajo, gasto enorme de recursos,

pérdida y omisiones en la información solicitada.

Actualmente este proceso lo llevan a cabo en tarjetas kardex, que para el efecto utilizan libros de Excel. Teniendo limitaciones cuando los datos empiezan a tener dimensiones de miles de registros; las relaciones entre los datos se hacen tediosas y cada vez se dificulta más el entendimiento de la masa de los datos.

Otra parte del proceso de seguimiento es que se llevan los datos en papel simple, y con la cantidad de información que manipulan, muchas veces se duplican, lo que conlleva a pérdida de tiempo, esfuerzo y recursos, por lo que resulta que el seguimiento de servicios que se dan en la Universidad no es adecuado.

El uso de un sistema informático facilitará el control de los servicios generales en las infraestructuras de la Universidad Técnica Estatal de Quevedo.

La mayor parte del proceso de seguimiento y control de datos se llevan en papel simple, y con la cantidad de información que manipulan, muchas veces se duplica la información, lo que conlleva la pérdida de tiempo, esfuerzo y recursos, por cuanto el seguimiento de servicios que se realiza para la Universidad no es adecuado, y peor aún eficiente.

Además el seguimiento oportuno de trabajos realizados no tiene un control adecuado de las actividades que realiza cada miembro del equipo de mantenimiento, de igual forma, no se lleva un registro de los trabajos realizados del equipo del personal de Servicios Generales.

Una vez identificadas las causas de los problemas encontrados muy frecuentemente en los procesos de desarrollo, se propone una solución mediante el desarrollo de un sistema informático que permita ingresar toda la información de cada proyecto, la asignación de recursos humanos al equipo de

desarrollo, tareas, actividades, etc.

El área de Servicios Generales de la Universidad Técnica Estatal de Quevedo no cuenta con un sistema que permita procesar los ingresos, egresos, informes, reportes, ni como saber con exactitud la existencia de material en bodega, de manera que si se desea realizar un informe semanal se tiene que buscar en forma manual lo que conlleva a la pérdida de tiempo y recurso humano, debido que este proceso lo llevan a cabo en tarjetas kardex, que para el efecto utilizan libros de Excel.

La importancia del sistema informático en el área de Servicios Generales de la Universidad Técnica Estatal de Quevedo, radica en cuanto al control de material, informes, egreso e ingreso se llevará de una manera tecnológica en dicha dependencia.

El impacto del sistema sobresale en el campo universitario, por el interés que hay de parte de quienes conforman el área de Servicios Generales de forma que pueden implementar este sistema para generar reporte de los datos oportunamente y organización en la bodega.

El beneficio que ofrece el sistema es que ayuda a los usuarios de dicha dependencia, por cuanto se les hará más fácil obtener la información sin pérdida de tiempo.

De manera que el sistema facilitará el trabajo de bodega en el área de Servicios Generales, además el área de Servicios Generales facilita toda la información necesaria y requerida para la elaboración del sistema.

El alcance del presente trabajo examina los siguientes puntos requeridos por la Unidad de Servicios Generales:

- Diseñar un modelo contextual, lógico y físico del aplicativo de gestión de

proyectos de software bajo la plataforma Visual Studio 2010.

- Módulo de reportes e informes que contenga los requerimientos necesarios para la U.S.G. (Unidad de Servicios Generales)
- Capacitación a usuarios, implantación del aplicativo y puesta en marcha del mismo.
- Diseñar y documentar el proceso de control de servicios e instalar las aplicaciones que permitirán llevar a cabo el proceso.

Implementar una estructura de procesos que cumpla con todas las tareas, requerimientos y mejoras prácticas tomadas en cuenta en la metodología Edraw Max, que sirva como marco de trabajo para el posterior desarrollo de proyectos dentro de la U.S.G.

1.1. Objetivos

1.1.1. General

Automatizar la gestión y el control de los servicios generales en la Universidad Técnica Estatal de Quevedo, mediante una aplicación informática.

1.1.2. Específicos

- Identificar los procesos de flujo de información que se llevan actualmente en la gestión y control de los servicios generales.
- Analizar las leyes, políticas y reglamentos, inmersos en la gestión y control de los bienes de las instituciones públicas y en particular de la Universidad Técnica Estatal de Quevedo.
- Diseñar interfaces amigables mediante el uso de herramienta de programación de cuarta generación.

1.2. Hipótesis

El Sistema Informático para la gestión y Control de la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo, mejorará los servicios a los usuarios y optimizará el trabajo de su personal.

CAPÍTULO II

MARCO TEÓRICO O MARCO REFERENCIAL

2.1. Sistema de Información

2.1.1. Que son los Sistemas de Información

Peña (2006). Un sistema de información es un conjunto de elementos interrelacionados con el propósito de prestar atención a las demandas de información de una organización, para elevar el nivel de conocimientos que permitan un mejor apoyo a la toma de decisiones y desarrollo de acciones.

Peralta (2008). Define sistema de información como: conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. Teniendo muy en cuenta el equipo computacional necesario para que el sistema de información pueda operar y el recurso humano que interactúa con el Sistema de Información, el cual está formado por las personas que utilizan el sistema.

Peralta (2008). Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información.

2.1.2. Entrada de Información

Peralta (2008). Es el proceso mediante el cual el Sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos. Esto último se denomina interfaces automáticas. Las unidades típicas de entrada de datos a las computadoras son las terminales, las cintas magnéticas, las unidades de diskette, los códigos de barras, los escáner, la voz, los monitores sensibles al tacto, el teclado y el mouse, entre otras.

2.1.3. Almacenamiento de información

Peralta (2008). El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos. La unidad típica de almacenamiento son los discos magnéticos o discos duros, los discos flexibles o diskettes y los discos compactos (CD-ROM).

2.1.4. Procesamiento de Información

Peralta (2008). Es la capacidad del Sistema de Información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere una proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.

2.1.5. Salida de Información

Peralta (2008). La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras, terminales, diskettes, cintas magnéticas, la voz, los graficadores y los plotters, entre otros. Es importante aclarar que la salida de un Sistema de Información puede constituir la entrada a otro Sistema de Información o módulo. En este caso, también existe una interface automática de salida.

Otro autor define que Un sistema de información es el sistema de personas, registros de datos y actividades que procesa los datos y la información en cierta organización, incluyendo manuales de procesos o procesos automatizados.

2.1.6. Tipos y Usos de los Sistemas de Información

Peralta (2008). Los Sistemas de Información que logran la automatización de procesos operativos dentro de una organización, son llamados frecuentemente Sistemas Transaccionales, ya que su función primordial consiste en procesar transacciones tales como pagos, cobros, pólizas, entradas, salidas, etc.

Por otra parte, los Sistemas de Información que apoyan el proceso de toma de decisiones son los Sistemas de Soporte a la Toma de Decisiones, Sistemas para la Toma de Decisión de Grupo, Sistemas Expertos de Soporte a la Toma de Decisiones y Sistema de Información para Ejecutivos. El tercer tipo de sistema, de acuerdo con su uso u objetivos que cumplen, es el de los Sistemas Estratégicos, los cuales se desarrollan en las organizaciones con el fin de lograr ventajas competitivas, a través del uso de la tecnología de información.

2.1.7. Sistemas Transaccionales

Peralta (2008). Sus principales características son:

- A través de éstos suelen lograrse ahorros significativos de mano de obra, debido a que automatizan tareas operativas de la organización.
- Con frecuencia son el primer tipo de Sistemas de Información que se implanta en las organizaciones. Se empieza apoyando las tareas a nivel operativo de la organización.
- Son intensivos en entrada y salida de información; sus cálculos y procesos suelen ser simples y poco sofisticados.

- Tienen la propiedad de ser recolectores de información, es decir, a través de estos sistemas se cargan las grandes bases de información para su explotación posterior.
- Son fáciles de justificar ante la dirección general, ya que sus beneficios son visibles y palpables.

2.1.8. Sistemas de Apoyo de las Decisiones

Peralta (2008). Suelen introducirse después de haber implantado los Sistemas Transaccionales más relevantes de la empresa, ya que estos últimos constituyen su plataforma de información.

La información que generan sirve de apoyo a los mandos intermedios y a la alta administración en el proceso de toma de decisiones.

Suelen ser intensivos en cálculos y escasos en entradas y salidas de información. Así, por ejemplo, un modelo de planeación financiera requiere poca información de entrada, genera poca información como resultado, pero puede realizar muchos cálculos durante su proceso.

No suelen ahorrar mano de obra. Debido a ello, la justificación económica para el desarrollo de estos sistemas es difícil, ya que no se conocen los ingresos del proyecto de inversión.

Suelen ser Sistemas de Información interactivos y amigables, con altos estándares de diseño gráfico y visual, ya que están dirigidos al usuario final.

Apoyan la toma de decisiones que, por su misma naturaleza son repetitivos y de decisiones no estructuradas que no suelen repetirse. Por ejemplo, un Sistema de Compra de Materiales que indique cuándo debe hacerse un pedido al proveedor o un Sistema de Simulación de Negocios que apoye la decisión de introducir un nuevo producto al mercado.

Estos sistemas pueden ser desarrollados directamente por el usuario final sin la participación operativa de los analistas y programadores del área de informática.

Este tipo de sistemas puede incluir la programación de la producción, compra de materiales, flujo de fondos, proyecciones financieras, modelos de simulación de negocios, modelos de inventarios, entre otros.

2.1.9. Sistemas Estratégicos

Peralta (2008). Su función primordial no es apoyar la automatización de procesos operativos ni proporcionar información para apoyar la toma de decisiones.

Suelen desarrollarse in house, es decir, dentro de la organización, por lo tanto no pueden adaptarse fácilmente a paquetes disponibles en el mercado.

Típicamente su forma de desarrollo es a base de incrementos y a través de su evolución dentro de la organización. Se inicia con un proceso o función en particular y a partir de ahí se van agregando nuevas funciones o procesos.

Su función es lograr ventajas que los competidores no posean, tales como ventajas en costos y servicios diferenciados con clientes y proveedores. En este contexto, los Sistema Estratégicos son creadores de barreras de entrada al negocio. Por ejemplo, el uso de cajeros automáticos en los bancos en un Sistema Estratégico, ya que brinda ventaja sobre un banco que no posee tal servicio. Si un banco nuevo decide abrir sus puertas al público, tendrá que dar este servicio para tener un nivel similar al de sus competidores.

Apoyan el proceso de innovación de productos y proceso dentro de la empresa debido a que buscan ventajas respecto a los competidores y una forma de hacerlo en innovando o creando productos y procesos.

Un ejemplo de estos Sistemas de Información dentro de la empresa puede ser un sistema MRP (Manufacturing Resource Planning) enfocado a reducir sustancialmente el desperdicio en el proceso productivo, o bien, un Centro de Información que proporcione todo tipo de información; como situación de créditos, embarques, tiempos de entrega, etc. En este contexto los ejemplos anteriores constituyen un Sistema de Información Estratégico si y sólo si, apoyan o dan forma a la estructura competitiva de la empresa.

Por último, es importante aclarar que algunos autores consideran un cuarto tipo de sistemas de información denominado Sistemas Personales de Información, el cual está enfocado a incrementar la productividad de sus usuarios.

2.1.10. Evolución de los Sistemas de Información

Peralta (2008). De la sección anterior se desprende la evolución que tienen los Sistemas de Información en las organizaciones. Con frecuencia se implantan en forma inicial los Sistemas Transaccionales y, posteriormente, se introducen los Sistemas de Apoyo a las Decisiones. Por último, se desarrollan los Sistemas Estratégicos que dan forma a la estructura competitiva de la empresa.

En la década de los setenta, Richard Nolan, un conocido autor y profesor de la Escuela de Negocios de la Universidad de Harvard, desarrolló una teoría que impactó el proceso de planeación de los recursos y las actividades de la informática.

Según Nolan, la función de la Informática en las organizaciones evoluciona a través de ciertas etapas de crecimiento, las cuales se explican a continuación:

Las aplicaciones típicas que se implantan son los Sistemas Transaccionales tales como nóminas o contabilidad.

El pequeño Departamento de Sistemas depende en la mayoría de los casos del área de contabilidad.

El tipo de administración empleada es escaso y la función de los sistemas suele ser manejada por un administrador que no posee una preparación formal en el área de computación.

El personal que labora en este pequeño departamento consta a lo sumo de un operador y/o un programador.

En esta etapa es importante estar consciente de la resistencia al cambio del personal y usuario (ciberfobia) que están involucrados en los primeros sistemas que se desarrollan, ya que estos sistemas son importantes en el ahorro de mano de obra.

Esta etapa termina con la implantación exitosa del primer Sistema de Información. Cabe recalcar que algunas organizaciones pueden vivir varias etapas de inicio en las que la resistencia al cambio por parte de los primeros usuarios involucrados aborta el intento de introducir el computador a la empresa.

2.1.11. Etapa de contagio o expansión

Peralta (2008). Los aspectos sobresalientes que permiten diagnosticar rápido que una empresa se encuentra en esta etapa son:

Fernández (2006). Se inicia con la implantación exitosa del primer Sistema de Información en la organización. Como consecuencia de lo anterior, el primer ejecutivo usuario se transforma en el paradigma o persona que se habrá que imitar.

Las aplicaciones que con frecuencia se implantan en esta etapa son el resto de los Sistemas Transaccionales no desarrollados en la etapa de inicio, tales como facturación, inventarios, control de pedidos de clientes y proveedores, cheques, entre otros.

El pequeño departamento es promovido a una categoría superior, donde depende de la Gerencia Administrativa o Contraloría.

El tipo de administración empleado está orientada hacia la venta de aplicaciones a todos los usuarios de la organización; en este punto suele contratarse a un especialista de la función con preparación académica en el área de sistemas.

Se inicia la contratación de personal especializado y nacen puestos tales como analista de sistemas, analista-programador, programador de sistemas, jefe de desarrollo, jefe de soporte técnico, entre otros cargos.

Las aplicaciones desarrolladas carecen de interfaces automáticas entre ellas, de tal forma que las salidas que produce un sistema se tienen que alimentar en forma manual a otro sistema, con la consecuente irritación de los usuarios.

Los gastos por concepto de sistemas empiezan a crecer en forma importante, lo que marca la pauta para iniciar la racionalización en el uso de los recursos computacionales dentro de la empresa. Este problema y el inicio de su solución marcan el paso a la siguiente etapa.

2.1.12. Etapa de control o formalización

Peralta (2008). Para identificar a una empresa que transita por esta etapa es necesario considerar los siguientes elementos:

Esta etapa de evolución de la Informática dentro de las empresas se inicia con la necesidad de controlar el uso de los recursos computacionales a través de las técnicas de presupuestación base cero (partiendo de que no se tienen nada) y la implantación de sistemas de cargos a usuarios (por el servicio que se presta).

Las aplicaciones están orientadas a facilitar el control de las operaciones del negocio para hacerlas más eficaces, tales como sistemas para control de flujo de fondos, control de órdenes de compra a proveedores, control de inventarios, control y manejo de proyectos, entre otros.

El departamento de sistemas de la empresa suele ubicarse en una posición gerencial, dependiendo del organigrama de la Dirección de Administración o Finanzas.

El tipo de administración empleado dentro del área de Informática se orienta al control administrativo y a la justificación económica de las aplicaciones a desarrollar. Nace la necesidad de establecer criterios para las prioridades en el desarrollo de nuevas aplicaciones. La cartera de aplicaciones pendientes por desarrollar empieza a crecer.

En esta etapa se inician el desarrollo y la implantación de estándares de trabajo dentro del departamento, tales como: estándares de documentación, control de proyectos, desarrollo y diseño de sistemas, auditoria de sistemas y programación.

Se integra a la organización del departamento de sistemas, personal con habilidades administrativas y preparadas técnicamente.

Se inicia el desarrollo de interfaces automáticas entre los diferentes sistemas.

2.1.13. Etapa de integración

Peralta (2008). Las características de esta etapa son las siguientes:

La integración de los datos y de los sistemas surge como un resultado directo de la centralización del departamento de sistemas bajo una sola estructura administrativa.

Las nuevas tecnologías relacionadas con base de datos, sistemas administradores de bases de datos y lenguajes de cuarta generación, hicieron posible la integración.

En esta etapa surge la primera hoja electrónica de cálculo comercial y los usuarios inician haciendo sus propias aplicaciones.

Esta herramienta ayudó mucho a que los usuarios hicieran su propio trabajo y no tuvieran que esperar a que sus propuestas de sistemas fueran cumplidas.

El costo del equipo y del software disminuyó por lo cual estuvo al alcance de más usuarios.

En forma paralela a los cambios tecnológicos, cambió el rol del usuario y del departamento de Sistemas de Información.

El departamento de sistemas evolucionó hacia una estructura descentralizada, permitiendo al usuario utilizar herramientas para el desarrollo de sistemas.

Los usuarios y el departamento de sistema iniciaron el desarrollo de nuevos sistemas, remplazando los sistemas antiguos, en beneficio de la organización.

2.1.14. Etapa de administración de datos

Peralta (2008). Entre las características que destacan en esta etapa están las siguientes:

El departamento de Sistemas de Información reconoce que la información es un recurso muy valioso que debe estar accesible para todos los usuarios.

Para poder cumplir con lo anterior resulta necesario administrar los datos en forma apropiada, es decir, almacenarlos y mantenerlos en forma adecuada para que los usuarios puedan utilizar y compartir este recurso.

El usuario de la información adquiere la responsabilidad de la integridad de la misma y debe manejar niveles de acceso diferentes.

2.1.15. Etapa de madurez

Peralta (2008). Entre los aspectos sobresalientes que indican que una empresa se encuentra en esta etapa, se incluyen los siguientes:

Al llegar a esta etapa, la Informática dentro de la organización se encuentra definida como una función básica y se ubica en los primeros niveles del organigrama (dirección).

Los sistemas que se desarrollan son Sistemas de Manufactura Integrados por Computadora, Sistemas Basados en el Conocimiento y Sistemas Expertos, Sistemas de Soporte a las Decisiones, Sistemas Estratégicos y, en general, aplicaciones que proporcionan información para las decisiones de alta administración y aplicaciones de carácter estratégico.

En esta etapa se tienen las aplicaciones desarrolladas en la tecnología de base de datos y se logra la integración de redes de comunicaciones con terminales en lugares remotos, a través del uso de recursos computacionales.

2.2. Administración

Ponce (2008). La administración se define como el proceso de diseñar y mantener un ambiente en el que las personas trabajando en grupo alcance con eficiencia metas seleccionadas. Esta se aplica a todo tipo de organizaciones bien sean pequeñas o grandes empresas lucrativas y no lucrativas, a las industrias manufactureras y a las de servicio.

En fin la administración consiste en darle forma, de manera consistente y constante a las organizaciones. Todas las organizaciones cuentan con personas que tienen el encargo de servirle para alcanzar sus metas, llamados Gerente, administradores etc.

Es el proceso que comienza con la visión del Nro 1 de la organización; la misión de la organización; fijar objetivos, las estrategias y políticas organizacionales, usando como herramienta el Mapa estratégico; todo esto teniendo en cuenta las fortalezas/debilidades de la organización y las oportunidades/amenazas del contexto (Análisis FODA). La planificación abarca el largo plazo (de 5 años a 10 o más años), el mediano plazo (entre 1 años y 5 años) y el corto plazo donde se desarrolla el presupuesto anual más detalladamente.

Comparándolo con los objetivos y metas fijados; se detectan los desvíos y se toman las medidas necesarias para corregirlos. El control se realiza a nivel estratégico, nivel táctico y a nivel operativo; la organización entera es evaluada, mediante un sistema de Control de gestión; por otro lado también se contratan auditorías externas, donde se analizan y controlan las diferentes áreas funcionales de la organización.

El objeto de estudio de la Administración son las organizaciones; por lo tanto es aplicable a Empresas privadas y públicas; Instituciones públicas y organismos estatales, y a las distintas instituciones privadas. Por ejemplo: iglesias; universidades; gobiernos y organismos municipales, provinciales, nacionales; hospitales; fundaciones, etc.; y a todos los tipos de empresas privadas; e incluso las familias y hogares.

La administración puede ser entendida como la disciplina que se encarga del manejo científico de los recursos y de la dirección del trabajo humano, enfocada a la satisfacción de un interés.

Stoner (2006). La administración es analizada, por ejemplo, por la teoría de las organizaciones, dedicada del estudio sobre la forma en que los seres humanos administran sus recursos (humanos, materiales e inmateriales) e interactúan con el medio externo para alcanzar sus objetivos.

Por otra parte, la administración es el conjunto de organismos a cargo de la aplicación de las directivas necesarias para el cumplimiento de las leyes y las normas.

Hay veces en que el término administración se utiliza para denominar a la ciencia social conocida como administración de empresas. Esta ciencia estudia la organización de las empresas y la forma en que gestionan los recursos, los procesos y los resultados de sus actividades.

2.2.1. ¿Qué es un inventario?

Rojo (2001). Un inventario es definible como aquel conjunto de bienes, tanto muebles como inmuebles, con los que cuenta una empresa para comerciar.

De este modo, con los elementos del inventario es posible realizar transacciones, tanto de compra como de venta, es posible someterlos a ciertos procesos de elaboración o modificación antes de comerciar con ellos.

Es posible encontrar varios tipos de inventarios, los que son clasificados según el rubro en el que opere la empresa.

Este se encuentra constituido por todos los bienes de la empresa, ya sean comerciales o mercantiles.

En este inventario deberán mostrarse todos los elementos que la empresa tiene disponibles para la venta. Si se cuenta con productos de características especiales y condiciones particulares, entonces, deberá constituirse una nueva lista que especifique a todos los elementos que entran en dicha categoría. Un ejemplo de esta situación, sería el caso de productos que ya han sido comprados, pero que aún no han sido recibidos por la empresa, así como también aquellos que se han entregado en consignación o aquellos que han sido utilizadas como un medio de pago a terceros.

En segundo lugar, un “Inventario de productos terminados” incluye todos los productos que una empresa industrial o manufacturera ha adquirido, y que deben ser modificados para encontrarse dispuestos a la venta. Otro tipo de inventario es el “Inventario de productos en proceso de fabricación”, y tal como su nombre lo indica, se trata del detalle de productos que se encuentran en pleno proceso de elaboración.

2.3. Base de Datos

Molina (2009). Las bases de datos son recursos que recopilan todo tipo de información, para atender las necesidades de un amplio grupo de usuarios. Su tipología es variada y se caracterizan por una alta estructuración y estandarización de la información.

2.3.1. Que son las bases de datos.

Molina (2009). Es el conjunto de informaciones almacenadas en un soporte legible por ordenador y organizadas internamente por registros (formado por todos los campos referidos a una entidad u objeto almacenado) y campos (cada uno de los elementos que componen un registro). Permite recuperar cualquier clase de información: referencias, documentos textuales, imágenes, datos estadísticos, etc.

Korth. (2000) Los sistemas de base de datos están diseñados para gestionar grandes bloques de información. La gestión de datos implica tanto la definición de estructuras para el almacenamiento de información como la provisión de mecanismos para la gestión de la información. Además los sistemas de base de datos deben mantener la seguridad de la información almacenada, pese a caídas del sistema o intentos de accesos no autorizados. Si los datos van a ser compartidos por varios usuarios, el sistema debe evitar posibles resultados anómalos.

2.3.2. Porque la creación de las bases de datos.

Álvarez (2009). Las bases de datos se ingeniaron gracias a la necesidad de las grandes empresas de almacenar ingentes cantidades de información de una forma rápida, sencilla y fiable, y que a su vez pudieran acceder a ella en cualquier momento sin necesidad de desplazarse a salas dedicadas a archivar documentación, como hasta hace poco se venía haciendo.

2.3.3. Mejora en la integridad de datos.

Marqués (2001). La integridad de la base de datos se refiere a la validez y la consistencia de los datos almacenados. Normalmente, la integridad se expresa mediante restricciones o reglas que no se pueden violar. Estas restricciones se pueden aplicar tanto a los datos, como a sus relaciones, y es el Sistema de Gestión de Base de Datos (SGBD) quien se debe encargar de mantenerlas.

2.3.4. Mejora en la seguridad.

Marqués (2001). La seguridad de la base de datos es la protección de la base de datos frente a usuarios no autorizados. Sin unas buenas medidas de seguridad, la integración de datos en los sistemas de bases de datos hace que éstos sean más vulnerables que en los sistemas de ficheros. Sin embargo, el Sistema de Gestión de Base de Datos (SGBD) permite mantener la seguridad mediante el establecimiento de claves para identificar al personal autorizado a utilizar la base de datos. Las autorizaciones se pueden realizar a nivel de operaciones, de modo que un usuario puede estar autorizado a consultar ciertos datos pero no a actualizarlos.

2.3.5. Abstracción de datos.

Korth. (2000). Un sistema de gestión de base de datos es una colección de archivos interrelacionados y un conjunto de programas que permiten a los usuarios acceder y modificar esos archivos. Un objetivo importante de un

sistema de base de datos es proporcionar a los usuarios una visión abstracta de los datos. Es decir el sistema esconde ciertos detalles de cómo se almacenan y mantienen los datos. Sin embargo para que el sistema sea manejable, los datos se deben extraer eficientemente.

2.3.6. Aumento de la concurrencia.

Marqués (2001). En algunos sistemas de ficheros, si hay varios usuarios que pueden acceder simultáneamente a un mismo fichero, es posible que el acceso interfiera entre ellos de modo que se pierda información o, incluso, que se pierda la integridad. La mayoría de los SGBD gestionan el acceso concurrente a la base de datos y garantizan que no ocurran problemas de este tipo.

2.3.7. Características.

Pérez (2007). Entre las principales características de los sistemas de base de datos podemos mencionar:

- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

2.3.8. Vulnerable a los fallos.

Pérez (2007). El hecho de que todo esté centralizado en el SGBD (Sistema de gestión de base de datos) hace que el sistema sea más vulnerable ante los fallos que puedan producirse. Es por ello que deben tenerse copias de seguridad (Backup).

2.3.9. Tipos de Campos.

Pérez (2007). Cada Sistema de Base de Datos posee tipos de campos que pueden ser similares o diferentes. Entre los más comunes podemos nombrar:

Numérico: entre los diferentes tipos de campos numéricos podemos encontrar enteros “sin decimales” y reales “decimales”.

Booleanos: poseen dos estados: Verdadero “Si” y Falso “No”.

Memos: son campos alfanuméricos de longitud ilimitada. Presentan el inconveniente de no poder ser indexados.

Fechas: almacenan fechas facilitando posteriormente su explotación. Almacenar fechas de esta forma posibilita ordenar los registros por fechas o calcular los días entre una fecha y otra.

Alfanuméricos: contienen cifras y letras. Presentan una longitud limitada (255 caracteres).

Autoincrementables: son campos numéricos enteros que incrementan en una unidad su valor para cada registro incorporado. Su utilidad resulta: Servir de identificador ya que resultan exclusivos de un registro.

2.3.10. Tipos de Base de Datos

Codd (2003). Las bases de datos pueden clasificarse de varias maneras, de acuerdo al contexto que se esté manejando, o la utilidad de la misma. Según la variabilidad de los datos almacenados

2.3.11. Bases de datos estáticas

Codd (2003). Éstas son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones.

2.3.12. Bases de datos dinámicas

Codd (2003). Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización, borrado y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de una tienda de abarrotes, una farmacia, un videoclub.

2.3.13. Bases de datos bibliográficas

Codd (2003). Solo contienen un subrogante (representante) de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero nunca el texto completo, porque si no, estaríamos en presencia de una base de datos a texto completo (o de fuentes primarias). Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

2.3.14. Bases de datos de texto completo

Codd (2003). Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas.

2.3.15. Bases de Datos o "bibliotecas" de información química o biológica

Codd (2003). Son bases de datos que almacenan diferentes tipos de información proveniente de la química, las ciencias de la vida o médicas. Se pueden considerar en varios subtipos:

- Las que almacenan secuencias de nucleótidos o proteínas.
- Las bases de datos de rutas metabólicas.
- Bases de datos de estructura, comprende los registros de datos experimentales sobre estructuras 3D de biomolecular.
- Bases de datos clínicas.
- Bases de datos bibliográficas (biológicas, químicas, médicas y de otros campos): PubChem, Medline, EBSCOhost.

2.3.16. Tipos de modelos de Datos

Raúl Uranga Cruz (2008). Existen fundamentalmente tres alternativas disponibles para diseñar las bases de datos: el modelo jerárquico, el modelo de red y el modelo relacional.

a) El modelo jerárquico

Uranga (2008). La forma de esquematizar la información se realiza a través de representaciones jerárquicas o relaciones de padre/hijo, de manera similar a la estructura de un árbol. Así, el modelo jerárquico puede representar dos tipos de relaciones entre los datos: relaciones de uno a uno y relaciones de uno a muchos.

En el primer tipo se dice que existe una relación de uno a uno si el padre de la estructura de información tiene un solo hijo y viceversa, si el hijo tiene solamente un padre. En el segundo tipo se dice que la relación es de uno a muchos si el padre tiene más de un hijo, aunque cada hijo tenga un solo padre.

b) Inconveniente del Modelo Jerárquico

Relación maestro-alumno, donde un maestro tiene varios alumnos, pero un alumno también tiene varios maestros, uno para cada clase. En este caso, si la información estuviera representada en forma jerárquica donde el padre es el maestro y el alumno es el hijo, la información del alumno tendrá que duplicarse para cada uno de los maestros.

Otra dificultad que presenta el modelo jerárquico de representación de datos es respecto a las bajas. En este caso, si se desea dar de baja a un padre, esto necesariamente implicará dar de baja a todos y cada uno de los hijos que dependen de este padre.

c) El modelo de red.

Uranga (2008). El modelo de red evita esta redundancia en la información, a través de la incorporación de un tipo de registro denominado el conector, que en este caso pueden ser las calificaciones que obtuvieron los alumnos de cada profesor.

La dificultad surge al manejar las conexiones o ligas entre los registros y sus correspondientes registros conectores.

d) El modelo relacional

Uranga (2008). Se está empleando con más frecuencia en la práctica, debido el rápido entendimiento por parte de los usuarios que no tienen conocimientos

profundos sobre Sistemas de Bases de Datos y a las ventajas que ofrece sobre los dos modelos anteriores.

En este modelo toda la información se representa a través de arreglos bidimensionales o tablas. Estas operaciones básicas son:

- Seleccionar renglones de alguna tabla (SELECT)
- Seleccionar columnas de alguna tabla (PROJECT)
- Unir o juntar información de varias tablas (JOIN)

Es importante mencionar que la mayoría de los paquetes que manejan bases de datos disponibles en el mercado poseen las instrucciones SELECT, PROJECT Y JOIN con diferentes nombres y modalidades.

2.4. SQL SERVER

2.4.1. Introducción a las Bases de Datos de SQL SERVER

Charte (2006). Cuando se crea una base de datos, se configura la estructura de almacenamiento de los datos. Esta estructura incluye al menos un archivo de datos y un archivo de registro de transacciones. Antes de crear una base de datos, debe entender cómo almacena los datos Microsoft SQL Server, así como la función del registro de transacciones.

2.4.2. Almacenamiento de los Datos

Solid (2007). Microsoft SQL Server es un sistema de gestión de base de datos relacional diseñado para almacenar datos organizados como entidades, con atributos sencillos y complejos así como comportamientos lógicos. Sin embargo, el formato y propósito de estos datos varían dependiendo de nuestros negocios en particular, su situación actual, su rendimiento pasado y como esperamos trabajar en el futuro. Cualesquiera que sean los datos que

decídanos almacenar, necesitarán registrarse, estructurarse y procesarse para que podamos encontrar respuestas a nuestras necesidades específicas de información. Almacenar simplemente los datos no es suficiente. Es necesaria una estructura para ayudarnos en la búsqueda de la información que necesitamos.

2.4.3. Archivos de Bases de Datos de SQL Server

Charte (2006). Las bases de datos de SQL Server se almacenan en archivos. Estos archivos se generan automáticamente cuando se crea una base de datos. Existe más de un tipo de archivo y cada base de datos utiliza siempre al menos dos archivos.

2.4.4. Tipos de archivos de las Bases de Datos

Charte (2006). Los archivos de bases de datos pueden ser:

Archivo de datos principal. Este archivo constituye el punto de partida de la base de datos. Todas las bases de datos deben tener un archivo de base de datos principal y sólo uno. Los nombres de archivos de datos principales normalmente tienen la extensión.

Archivos de registro. Estos archivos contienen toda la información de registro de transacciones que se utiliza para recuperar la base de datos. Todas las bases de datos tienen al menos un archivo de registro. Los nombres de archivos de registro normalmente tienen la extensión.

2.4.5. Componentes del SQL

Casares (2004). El lenguaje **SQL** está compuesto por comandos, cláusulas, operadores y funciones de agregado. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos.

2.4.5.1. Comandos

Comandos DLL

Los DLL que permiten crear y definir nuevas bases de datos, campos e índices.

Comando	Descripción
CREATE	Utilizado para crear nuevas tablas, campos e índices.
DROP	Empleado para eliminar tablas e índices.
ALTER	Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos.

Cuadro 1 Comandos DLL

Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

Comandos DML

Los DML que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.

Comando	Descripción
SELECT	Utilizado para consultar registros de la base de datos que satisfagan un criterio determinado.
INSERT	Utilizado para cargar lotes de datos en la base de datos en una única operación.
UPDATE	Utilizado para modificar los valores de los campos y registros especificados.
DELETE	Utilizado para eliminar registros de una tabla de una base de datos.

Cuadro 2 Comandos DML

Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

2.4.5.2. Cláusulas

Las cláusulas son condiciones de modificación utilizadas para definir los datos que desea seleccionar o manipular.

Comando	Descripción
FROM	Utilizada para especificar la tabla de la cual se van a seleccionar los registros
WHERE	Utilizada para especificar las condiciones que deben reunir los registros que se van a seleccionar
GROUP BY	Utilizada para separar los registros seleccionados en grupos específicos
HAVING	Utilizada para expresar la condición que debe satisfacer cada grupo
ORDER BY	Utilizada para ordenar los registros seleccionados de acuerdo con un orden específico

Cuadro 3 Clausulas

Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

2.4.5.3. Operadores Lógicos

Los operadores lógicos sirven para combinar condiciones.

Operador	Uso
AND	Es el “y” lógico. Evalúa dos condiciones y devuelve un valor de verdad sólo si ambas son ciertas.
OR	Es el “o” lógico. Evalúa dos condiciones y devuelve un valor de verdad si alguna de las dos es cierta.
NOT	Negación lógica. Devuelve el valor contrario de la expresión.

Cuadro 4 Operadores Lógicos

Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

2.4.5.4. Operadores de Comparación

Un operador de la comparación compara sus operando y devuelve un valor lógico basado en si la comparación es verdad o no.

Operador	Uso
<	Menor que
>	Mayor que
<>	Distinto de
<=	Menor ó Igual que
>=	Mayor ó Igual que
BETWEEN	Utilizado para especificar un intervalo de valores.
LIKE	Utilizado en la comparación de un modelo
In	Utilizado para especificar registros de una base de datos

Cuadro 5 Operadores de Comparación

Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

2.4.5.5. Funciones de Agregado

Las funciones de agregado se usan dentro de una cláusula **SELECT** en grupos de registros para devolver un único valor que se aplica a un grupo de registros.

Comando	Descripción
AVG	Utilizada para calcular el promedio de los valores de un campo determinado
COUNT	Utilizada para devolver el número de registros de la selección
SUM	Utilizada para devolver la suma de todos los valores de un campo determinado
MAX	Utilizada para devolver el valor más alto de un campo especificado
MIN	Utilizada para devolver el valor más bajo de un campo especificado

Cuadro 6 Funciones de Agregado

Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

2.5. Visual Studio .Net

2.5.1. Introducción a Visual Studio .NET

Charte (2006). Visual Studio .NET es la Herramienta Rápida de Desarrollo (RAD) de Microsoft para la siguiente generación de Internet que son los Servicios Web XML. Esta herramienta permite la creación de aplicaciones usando el Marco .NET, es decir usando el CLR, la Librería de Clases, ADO .NET, ASP .NET, etc.

Es un software que brinda las herramientas necesarias para crear, distribuir, administrar y dar mantenimiento a aplicaciones Web distribuidas que usan Servicios Web XML, todo esto con una gran facilidad, rapidez y bajo costo.

Se puede crear aplicaciones Web directamente usando el Framework .NET y algún programa editor, por ejemplo el Bloc de Notas, pero el tiempo que llevaría el desarrollo no justificaría el ahorro de costos, en cambio, si se utiliza una herramienta como Visual Studio .NET el tiempo de desarrollo se reduciría enormemente.

Visual Studio .NET permite también la integración y el uso cruzado de lenguajes de programación: Visual Basic .NET, Visual C# .NET, Visual C++ .NET y JScript .NET

A diferencia de la versión anterior no existe Visual InterDev, ni Visual J++, además Visual FoxPro .NET no comparte las características unificadas del Marco .NET

Las aplicaciones creadas en la plataforma .net podrán ser usadas por un gran número de clientes, tales como PC, PC de bolsillo (Palm PC), celulares digitales, PC de juegos (PC Games), PC de auto (Auto PC), PC Televisor (Ultimate TV), etc.

2.5.2. Microsoft .Net Framework

Charte (2006). Lo que se dice en el eBook Microsoft .NET Framework. ".NET Framework es un entorno para construir, instalar y ejecutar servicios Web y otras aplicaciones. Se compone de tres partes principales: el Common Language Runtime, las clases Framework y ASP.NET"

Lo que dice el MSDN Library.

El .NET Framework es un entorno multi-lenguaje para la construcción, distribución y ejecución de Servicios Webs y aplicaciones."

"El .NET Framework es una nueva plataforma diseñada para simplificar el desarrollo de aplicaciones en el entorno distribuido de Internet."

El .NET Framework es el corazón de .NET, cualquier cosa que queramos hacer en cualquier lenguaje .NET debe pasar por el filtro cualquiera de las partes integrantes del .NET Framework.

2.5.3. La Plataforma Microsoft .Net

Charte (2006). La plataforma Microsoft .NET es un conjunto de tecnologías diseñadas para transformar el Internet, a que este sea completamente escalable. Esto proporciona nuevas formas de construir aplicaciones a partir de las colecciones de Servicios Web.

La plataforma Microsoft .NET está compuesta por:

- El Marco .Net
- Los Bloques de Servicios .Net
- Visual Studio .Net
- Los Servidores Empresariales .Net

2.5.4. Marco .Net

Charte (2006). Es la parte central de la plataforma Microsoft .Net y consiste en un compilador común llamado CLR (Common Language Runtime).

2.5.5. Componentes del Marco .Net

- ASP .Net: Formularios Web Servicios Web
- Windows Form: Interfaz de usuario
- ADO .Net: Datos y XML
- Librería de clases del Marco .Net

2.5.6. Common Language Runtime (CLR)

Charte (2006). Conocido como el lenguaje común en tiempo de ejecución. Es el Runtime común que soportan todos los lenguajes del Marco .Net y que realiza la mayor parte de las funciones para que corran las aplicaciones .Net.

El Common Language Runtime (CLR) es una serie de librerías dinámicas (DLLs), también llamadas assemblies, que hacen las veces de las DLLs del API de Windows así como las librerías runtime de Visual Basic o C++. Cualquier ejecutable depende de una forma u otra de una serie de librerías, ya sea en tiempo de ejecución como a la hora de la compilación. Pues el CLR es eso, una serie de librerías usadas en tiempo de ejecución para que nuestros ejecutables o cualquiera basado en .NET puedan funcionar. Se acabó eso de que existan dos tipos de ejecutables: los que son autosuficientes y no dependen de librerías externas o los que necesitan de librerías en tiempo de ejecución para poder funcionar, tal es el caso de las versiones anteriores de Visual Basic. Por otro lado, la librería de clases de .NET Framework proporciona una jerarquía de clases orientadas a objeto disponibles para cualquiera de los lenguajes basados en .NET, incluido el Visual Basic. También permite crear Threads o hilos o tramas de ejecución.

2.5.7. Tipos de Ejecutables

Charte (2006). Con Visual Basic .NET puedes crear básicamente estos dos tipos de ejecutables: de consola, no gráfico, al estilo del viejo MS-DOS, y gráficos, como los que normalmente estamos acostumbrados a ver en Windows.

Existen otros tipos de aplicaciones que se pueden crear con Visual Basic .NET: aplicaciones ASP.NET, (realmente no es una aplicación o ejecutable, sino un compendio de distintos tipos de elementos...), servicios Web, servicios Windows, etc.

2.5.8. ¿Qué es un Namespace (o espacio de nombres)?

Charte (2006). Un espacio de nombres es un esquema lógico de nombres para tipos en el que un nombre de tipo simple, como Mi Tipo, aparece precedido por un nombre jerárquico separado por puntos. [...]"

Un Namespace, es una forma de agrupar clases, funciones, tipos de datos, etc. que están relacionadas entre sí. Por ejemplo, entre los Namespaces que podemos encontrar en el .NET Framework encontramos uno con funciones relacionadas con Visual Basic: Microsoft. Visual Basic. Si te fijas, Microsoft y Visual Basic están separados por un punto, esto significa que Microsoft a su vez es un Namespace que contiene otros "espacios de nombres", tales como el mencionado Visual Basic, CSharp y Win32 con el cual podemos acceder a eventos o manipular el registro del sistema.

Por regla general se deberían agrupar en un Namespace funciones o clases que estén relacionadas entre sí. De esta forma, será más fácil saber que estamos trabajando con funciones relacionadas entre sí.

2.5.9. ¿Qué es un Assembly (o ensamblado)?

Charte (2006). Un ensamblado es el bloque constructivo primario de una aplicación de .NET Framework. Se trata de una recopilación de funcionalidad que se construye, versiona e instala como una única unidad de implementación (como uno o más archivos). [...]"

Un Assembly es una librería dinámica (DLL) en la cual pueden existir distintos espacios de nombres.

Un ensamblado o Assembly puede estar formado por varios ficheros DLLs y EXEs, pero lo más importante es que todos los ensamblados contienen un manifiesto (o manifest), gracias al cual se evitan muchos de los quebraderos de cabeza a los que Windows nos tiene acostumbrados, al menos en lo referente a las distintas versiones de las librerías y ejecutables, seguramente habrás oído hablar de las DLL Hell (o librerías del demonio) expresión que se usa cuando hay incompatibilidad de versiones entre varias librerías que están relacionadas entre sí.

2.5.10. Librerías de Clases del Marco .Net

Charte (2006). Conjunto de clases comunes. Unifica los diferentes marcos existentes: APIs, MFC, WFC, VB y VBA. La clase base principal se llama System.

ADO .Net: Datos y XML

ADO .Net proporciona un amplio soporte XML.

ASP .Net: Formularios y Servicios Web

ASP .Net es un marco de programación, incorporable en el CLR y puede construir aplicaciones Web potentes.

2.5.11. Creación de Componentes .NET

Charle (2006). Crear componentes o controles de usuario es tan simple como crear un formulario, ya que usando la herencia se puede pasar todas las características de un objeto a otro, esto está presente en todos los objetos creados en Visual Studio .NET, sean visuales o no.

2.5.12. Creación de Archivos XML

Charle (2006). Con el diseñador de XML, crear un archivo XML es más fácil que nunca, ya que se muestra de colores el código y se auto completan los Tags que uno va escribiendo. Este maneja 3 vistas: XML, esquemas y datos.

2.5.13. Lenguajes en .NET

Charle (2006). En Visual Studio .NET vienen los siguientes Lenguajes de Programación:

- Visual Basic .NET
- Visual C# .NET
- Visual C++ .NET
- Visual JScript .NET.

2.6. Ingeniería de Software

2.6.1. ¿Qué es la Ingeniería de Software?

Pressman (2007). Es la disciplina o área de la Ingeniería que ofrece métodos y técnicas para desarrollar y mantener software. La creación del software es un proceso intrínsecamente creativo y la Ingeniería del Software trata de sistematizar este proceso con el fin de acotar el riesgo del fracaso en la consecución del objetivo creativo por medio de diversas técnicas que se han

demostrado adecuadas en base a la experiencia previa. Esta ingeniería trata con áreas muy diversas de la informática y de las ciencias de la computación, tales como construcción de compiladores, sistemas operativos, o desarrollos Intranet/Internet, abordando todas las fases del ciclo de vida del desarrollo de cualquier tipo de sistemas de información y aplicables a infinidad de áreas.

2.6.2. Definición de Ingeniería de Software

Pressman (2007). La Ingeniería del Software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar y operar (funcionar) y mantenerlos. Así como también desarrollo de software o producción de software.

La Ingeniería del Software es el establecimiento y uso de principios sólidos de la ingeniería para obtener económicamente un software confiable y que funcione de modo eficiente en máquinas reales.

2.6.3. Calidad del Software

Weitzenfeld (2008). La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario. Sin embargo la calidad del software es un concepto complejo que no es directamente comparable con la calidad de la manufactura de producto. Los productos de software se han convertido hoy en día en uno de los principales objetivos estratégicos de las organizaciones debido a que, cada vez más, los procesos más importantes de las organizaciones y por lo tanto su supervivencia depende del buen funcionamiento de los sistemas de software.

Calidad, significa desarrollar, diseñar y producir y mantener un producto que sea el más económico, el más útil y siempre satisfactorio para el consumidor.

2.6.4. Definición: Ingeniería

Quispe-Otazu (2007). La ingeniería es el estudio y la aplicación de las distintas ramas de la tecnología. El profesional en este ámbito recibe el nombre de ingeniero.

La actividad del ingeniero supone la concreción de una idea en la realidad. Esto quiere decir que, a través de técnicas, diseños y modelos, y con el conocimiento proveniente de las ciencias, la ingeniería puede resolver problemas y satisfacer necesidades humanas.

La ingeniería también supone la aplicación de la inventiva y del ingenio para desarrollar una cierta actividad. Esto, por supuesto, no implica que no se utilice el método científico para llevar a cabo los planes.

2.6.5. Definición de Software

Quispe-Otazu (2007). Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

El software no son solo programas, sino todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta.

Un sistema de software consiste en diversos programas independientes, archivos de configuración que se utilizan para ejecutar estos programas, un sistema de documentación que describe la estructura del sistema, la documentación para el usuario que explica cómo utilizar el sistema y sitios web que permitan a los usuarios descargar la información de productos recientes.

El software de computadora es el producto que los ingenieros de software construyen y después mantienen en el largo plazo, el software se forma con:

- Las instrucciones (programas de computadora) que al ejecutar se proporcionan las características, funciones y el grado de desempeño deseados.
- Las estructuras de datos que permiten que los programas manipulen información de manera adecuada.
- Los documentos que describen la operación y uso de los programas.

2.6.6. Análisis de Sistemas.

Kenneth (2009). El Análisis de Sistemas trata básicamente de determinar los objetivos y límites del sistema objeto de análisis, caracterizar su estructura y funcionamiento, marcar las directrices que permitan alcanzar los objetivos propuestos y evaluar sus consecuencias. Dependiendo de los objetivos del análisis, podemos encontrarnos ante dos problemáticas distintas:

- Análisis de un sistema ya existente para comprender, mejorar, ajustar y/o predecir su comportamiento
- Análisis como paso previo al diseño de un nuevo sistema-producto

En cualquier caso, podemos agrupar más formalmente las tareas que constituyen el análisis en una serie de etapas que se suceden de forma iterativa hasta validar el proceso completo:

- **Conceptualización.**

Consiste en obtener una visión de muy alto nivel del sistema, identificando sus elementos básicos y las relaciones de éstos entre sí y con el entorno.

- **Análisis funcional.**

Describe las acciones o transformaciones que tienen lugar en el sistema.

Dichas acciones o transformaciones se especifican en forma de procesos que reciben unas entradas y producen unas salidas.

- **Análisis de condiciones** (o constricciones).

Debe reflejar todas aquellas limitaciones impuestas al sistema que restringen el margen de las soluciones posibles. Estas se derivan a veces de los propios objetivos del sistema:

- Operativas, como son las restricciones físicas, ambientales, de mantenimiento, de personal, de seguridad, etc.
- De calidad, como fiabilidad, mantenibilidad, seguridad, convivencia, generalidad, etc.

Sin embargo, en otras ocasiones las constricciones vienen impuestas por limitaciones en los diferentes recursos utilizables:

- Económicos, reflejados en un presupuesto
- Temporales, que suponen unos plazos a cumplir
- Humanos
- Metodológicos, que conllevan la utilización de técnicas determinadas
- Materiales, como espacio, herramientas disponibles, etc.

- **Construcción de modelos.**

Kenneth (2009). Una de las formas más habituales y convenientes de analizar un sistema consiste en construir un prototipo (un modelo en definitiva) del mismo.

- **Validación del análisis.**

Kenneth (2009). A fin de comprobar que el análisis efectuado es correcto y evitar, en su caso, la posible propagación de errores a la fase de diseño, es

imprescindible proceder a la validación del mismo. Para ello hay que comprobar los extremos siguientes:

- El análisis debe ser consistente y completo.
- Si el análisis se plantea como un paso previo para realizar un diseño, habrá que comprobar además que los objetivos propuestos son correctos y realizables.

Una ventaja fundamental que presenta la construcción de prototipos desde el punto de vista de la validación radica en que estos modelos, una vez contruidos, pueden ser evaluados directamente por los usuarios o expertos en el dominio del sistema para validar sobre ellos el análisis.

2.6.7. Diseño de Sistemas.

Kenneth (2009). El Diseño de Sistemas se ocupa de desarrollar las directrices propuestas durante el análisis en función de aquella configuración que tenga más posibilidades de satisfacer los objetivos planteados tanto desde el punto de vista funcional como del no funcional (lo que antes hemos denominado constricciones). El proceso de diseño de un sistema complejo se suele realizar de forma descendente:

- Diseño de alto nivel (o descomposición del sistema a diseñar en subsistemas menos complejos).
- Diseño e implementación de cada uno de los subsistemas:
 - Especificación consistente y completa del subsistema de acuerdo con los objetivos establecidos en el análisis.
 - Desarrollo según la especificación.
 - Prueba.
- Integración de todos los subsistemas.
- Validación del diseño.

Dentro del proceso de diseño de sistemas hay que tener en cuenta los efectos que pueda producir la introducción del nuevo sistema sobre el entorno en el que deba funcionar, adecuando los criterios de diseño a las características del mismo.

De estas cuestiones se ocupa una disciplina, la Ergonomía, que tiene por objeto la optimización de los entornos hombre-máquina. Si bien en un principio estaba centrada en los aspectos antropométricos de la relación hombre-máquina, en la actualidad ha pasado a intervenir con fuerza en todos los procesos cognitivos (análisis, interpretación, decisión, comunicación y representación del conocimiento). Así, con respecto al diseño de herramientas software, la ergonomía tiene mucho que decir en cuestiones relacionadas con la disposición de informaciones en pantalla, profundidad de menús, formato de iconos, nombres de comandos, control de cursores, tiempos de respuesta, manejo de errores, estructuras de datos, utilización de lenguaje natural, etc.

2.6.8. Introducción

Rerych (2007). Una metodología de desarrollo de software se refiere a un framework que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información.

A lo largo del tiempo, una gran cantidad de métodos han sido desarrollados diferenciándose por su fortaleza y debilidad.

El framework para metodología de desarrollo de software consiste en:

- Una filosofía de desarrollo de software con el enfoque del proceso de desarrollo de software
- Herramientas, modelos y métodos para asistir al proceso de desarrollo de software

Estos frameworks son a menudo vinculados a algún tipo de organización, que además desarrolla, apoya el uso y promueve la metodología. La metodología es a menudo documentada en algún tipo de documentación formal.

2.6.9. Enfoques de desarrollo de software

Rerych (2007). Cada metodología de desarrollo de software tiene más o menos su propio enfoque para el desarrollo de software. Estos son los enfoques más generales, que se desarrollan en varias metodologías específicas. Estos enfoques son los siguientes:

- Modelo en cascada: Framework lineal.
- Prototipado: Framework iterativo.
- Incremental: Combinación de framework lineal e iterativo.
- Espiral: Combinación de framework lineal e iterativo.
- RAD: Rapid Application Development, framework iterativo.

2.6.10. Metodología de desarrollo OPEN UP

Eclipse (2008). OPEN UP es un proceso que es la versión libre o corta del RUP. Este plantea que se debe tener una versión ejecutable del proyecto en poco tiempo. Para ello este modelo plantea que solo se debe usar los procesos que sean necesarios. Y también no se debe usar muchos artefactos, además debe acoplarse a las necesidades del usuario pudiendo ser modificado y extendido. Usa las mismas fases e hitos del RUP pero sin tanta documentación. Se enfoca en el valor del software y disminuye riesgos. Este modelo sirve tanto para proyectos pequeños como para proyectos grandes. En mi opinión este modelo garantiza calidad a pesar de que para su desarrollo no necesite mucho tiempo ni tanta documentación además que disminuye riesgos y puede ser utilizado en gran variedad de proyectos.

OPEN UP es una metodología de desarrollo basada en la metodología RUP (Rational Unified Process). Es el subconjunto de esta última que contiene el conjunto mínimo de prácticas que ayuden a un equipo de desarrollo software a realizar un producto de alta calidad y de una forma más eficiente. Open Up utiliza un punto de vista pragmático y una filosofía ágil que se centraliza en la naturaleza colaborativa del proceso de desarrollo del software. Una de sus principales características es su alto grado de adaptabilidad a las necesidades de un proyecto en particular.

Eclipse (2008). OPEN UP intenta incluir dentro de su proceso de desarrollo únicamente el contenido imprescindible para garantizar un proceso de desarrollo de calidad y eficiente. Por este motivo no incluye guías sobre aspectos que suelen aparecer en la gestión de proyectos como puedan ser equipos de desarrollo grandes, negociaciones contractuales, uso de tecnologías específicas, definición de aplicaciones de seguridad o críticas, etc. Esto no quiere decir que Open UP no contemple estos aspectos, simplemente los deja sin especificar, pero en cualquier momento se puede extender la metodología para cubrir alguno de estos aspectos.

Open Up tiene las características de un Unified Process al cual se aplican enfoques iterativos e incrementales dentro de un ciclo de vida estructurado. El proceso de desarrollo en OpenUp utiliza casos de uso, escenarios, gestión del riesgo y un enfoque centralizado en la arquitectura.

2.6.11. Principios de OPEN UP

Los cuatro principios básicos en los que se fundamenta OpenUp son los siguientes:

2.6.11.1. Promover la colaboración para alinear los intereses comunes y difundir el conocimiento.

Este principio promueve la colaboración, desarrollo y compartición de un conocimiento sobre el proyecto en toda la organización, así como un entorno de trabajo saludable.

2.6.11.2. Balanceo de las prioridades competitivas para maximizar el valor de los participantes del proyecto.

Se intentan promover prácticas que permitan a los participantes del proyecto el desarrollo de una solución que maximice sus beneficios aportando el máximo valor de negocio.

2.6.11.3. Desarrollo de una arquitectura al principio con el fin de minimizar riesgos y planificar el desarrollo.

Ejercer prácticas que permitan al equipo focalizarse en la arquitectura del sistema con el fin de reducir riesgos y que permitan planificar la estrategia de desarrollo.

2.6.11.4. Evolucionar para obtener un feedback continuo en aras de una mejora continua.

Con este principio se intentan ejecutar prácticas que permitan obtener un feedback desde el inicio del proyecto y de forma continua, así como mostrar el valor de negocio a los participantes del proyecto.

Eclipse (2008). Disciplinas: OpenUp se centra en las siguientes disciplinas: Requisitos, arquitectura, desarrollo, test, gestión de proyecto, gestión de la configuración y del cambio. Como se puede observar, hay ciertas disciplinas de RUP que han sido omitidas, como modelización del negocio, definición del

entorno, gestión de requisitos avanzada y configuración de herramientas para la gestión de la configuración. Estas disciplinas no se incluyeron en OpenUp porque fueron consideradas no necesarias para proyectos de pequeña magnitud o se sabe que serían manejadas por organizaciones ajenas al equipo de desarrollo.

Tareas: En OpenUp se define tarea como la unidad de trabajo que debe ser realizada por algún rol. En OpenUp se cuenta con un total de 18 tareas que son ejecutadas de forma principal por un rol (el encargado de realizar la tarea) y de forma secundaria por roles que apoyan en la ejecución de dicha tarea. La naturaleza colaborativa de OpenUp queda demostrada al observarse que ninguna tarea es ejecutada de forma aislada por un único rol sin el apoyo de ningún otro.

Artefactos: Un artefacto en OpenUp se considera a todo aquello que una tarea necesita para realizar su función, o bien lo produce o modifica. Los distintos roles existentes son los encargados de crear y actualizar los artefactos.

Eclipse (2008). En OpenUp existen un total de 17 artefactos predefinidos considerados como esenciales, pero lógicamente la organización puede decidir añadir tantos artefactos propios como considere necesario, incluso se puede optar por ignorar estos artefactos predefinidos y utilizar cualquier tipo de artefactos definidos por la organización, siempre y cuando estos cumplan el objetivo para el cual fueron creados: almacenar información. Generalmente los artefactos suelen estar regulados por un sistema de control de versiones.

Procesos: Antes mencionamos que el contenido metodológico provee de explicaciones detalladas que describen como obtener ciertos objetivos independientemente de cómo se coloquen los elementos metodológicos dentro de un ciclo de vida. Los procesos toman los elementos metodológicos y los relacionan entre sí dentro de secuencias temporales que satisfacen las necesidades de distintos tipos de proyecto. Aquellos elementos metodológicos

con un alto grado de reutilización se agrupan en patrones de capacitación, aportando un enfoque de desarrollo consistente para la mayoría de proyectos de desarrollo. Estos patrones se realizan organizando las tareas en actividades, agrupándolas dentro de una secuencia que tiene sentido para un área en particular en la que el patrón es aplicado. Algunos de estos proyectos pueden usarse como plantillas para alguna de las iteraciones del proyecto.

2.6.11.5. Ciclo de vida de OpenUP

Eclipse (2008). Todo proyecto en OpenUp consta de cuatro fases: Concepción, Elaboración, Construcción y Transición. Cada una de estas fases se divide a su vez en iteraciones cada una de las cuales tiene como objetivo la entrega de un software operativo y funcional.

Fase de concepción: Las necesidades de cada participante del proyecto son tenidas en cuenta y son plasmadas en objetivos del proyecto. Se deben definir el ámbito del proyecto, los límites del mismo y el criterio de aceptación del proyecto. Los casos de uso críticos, aquellos que dirigen la funcionalidad del sistema, son definidos en esta fase, así como una estimación inicial del coste del proyecto y un boceto de la planificación.

Fase de elaboración: En esta fase se realizan tareas de análisis del dominio y definición de la arquitectura del sistema. Si se decide continuar con el proyecto se debe elaborar un plan de proyecto en esta fase, para lo cual se deben establecer unos requisitos y arquitectura estables. Por otro lado el proceso de desarrollo, las herramientas, la infraestructura a utilizar y el entorno de desarrollo también se especifican en detalle en esta fase. Al final de la fase se debe tener una definición clara y precisa de los casos de uso, los actores, la arquitectura del sistema y un prototipo ejecutable de la misma.

Fase de construcción: Todos los componentes y funcionalidades del sistema que falten por implementar son realizados, testeados e integrados en esta fase.

Los resultados obtenidos en forma de incrementos ejecutables deben ser desarrollados de la forma más rápida posible sin dejar de lado la calidad de lo desarrollado.

Fase de transición: Cuando el producto está lo suficientemente maduro (algo que es establecido por ejemplo en función del número de peticiones de cambio por parte de los clientes) como para ser introducido en la comunidad de usuarios, el proyecto se encuentra en esta fase. Las fases de la transición constan de subfases de testeo de versiones beta, pilotaje y capacitación de los usuarios finales y de los encargados del mantenimiento del sistema. En función de la respuesta obtenida por los usuarios puede que haya que realizar cambios en las entregas finales o implementar alguna funcionalidad más.

2.6.11.6. Prácticas de OpenUP

Eclipse (2008). OpenUp es una metodología basada en RUP, y por lo tanto, comparte las mismas prácticas que subyacen por debajo del flujo de trabajo y los roles de OpenUp. Estas prácticas son:

Desarrollar software de manera iterativa: Todo desarrollo debe realizarse de modo iterativo y en pequeños incrementos. Esto permite identificar riesgos de forma prematura, y actuar en consecuencia.

Gestionar los requisitos: La identificación de los requisitos clave y los que cambian continuamente es un proceso de que debe ejecutarse continuamente. Deben articularse los medios que permitan priorizar, filtrar y hacer un seguimiento de los requisitos, ya que, la comunicación será mucho más eficaz si está basada en unos requisitos claramente definidos.

Hacer uso de una arquitectura basada en componentes: Las arquitecturas basadas en componentes son más flexibles, y dado que OpenUp es una metodología ágil, y por lo tanto flexible, la arquitectura definida debe estar

basada en componentes. Por otro lado, la construcción de componentes con un alto grado de reutilización puede ahorrar considerable esfuerzo de futuros desarrollos.

Modelizar software de forma visual: Se deben aplicar métodos de visualización de modelos que permitan entender la complejidad del sistema sin dar cabida a la ambigüedad.

Verificar la calidad del software: Si se realizan verificaciones en cada iteración de cada una de las fases, se reduce considerablemente el costo de resolver cualquier defecto, ya que estos se identifican prematuramente.

Controlar los cambios aplicados al software: Todo cambio que se solicite sobre los requisitos establecidos se debe gestionar y realizar un seguimiento sobre el mismo, de este modo se puede determinar el grado de madurez del software desarrollado.

2.6.11.7. Roles de OpenUP

Eclipse (2008). Los roles de OpenUp que recogen el conjunto esencial de habilidades necesarias para realizar los desarrollos son los siguientes:

Cliente: Representan el conjunto de individuos o entidades cuyas necesidades se verán cubiertas por el proyecto. Cualquier individuo que se vea afectado por el resultado del proyecto puede adoptar este rol.

Analista: Dentro del equipo es el encargado de representar al cliente y sus necesidades. Es el encargado de recoger las impresiones de los clientes con el fin de entender el problema y establecer las prioridades de los requisitos identificados.

Arquitecto: Es el encargado de realizar la arquitectura general del sistema y tomar las decisiones técnicas claves que influenciarán en gran medida el futuro diseño e implementación del proyecto.

Desarrollador: El diseño, implementación, testeo unitario e integración de una parte del sistema es una tarea llevada a cabo por este rol.

Testeador: Como su nombre indica, es el rol encargado de identificar, diseñar e implementar y ejecutar los tests necesarios, además de dejar registro de los resultados obtenidos en dichos tests.

Gestor del proyecto: Es el encargado de dirigir la planificación del proyecto en colaboración con el equipo y los clientes. Asimismo debe conseguir que el equipo se mantenga focalizado en cumplir los objetivos planificados.

Cualquier rol: Es un rol que indica cualquier integrante del equipo que sea capaz de realizar tareas generales.

2.6.11.8. Otros enfoques de desarrollo de software

Garzón (2005). Metodologías de desarrollo Orientado a objetos, Diseño orientado a objetos (OOD) de Grady Booch, también conocido como Análisis y Diseño Orientado a Objetos (OOAD). El modelo incluye seis diagramas: de clase, objeto, estado de transición, la interacción, módulo, y el proceso.

- Top-down programming, evolucionado en la década de 1970 por el investigador de IBM Harlan Mills (y NiklausWirth) en Desarrollo Estructurado.
- Proceso Unificado, es una metodología de desarrollo de software, basado en UML. Organiza el desarrollo de software en cuatro fases, cada una de ellas con la ejecución de una o más iteraciones de desarrollo de software: creación, elaboración, construcción, y las directrices. Hay una serie de herramientas y productos diseñados para facilitar la aplicación. Una de las versiones más populares es la de Rational Unified Process.

2.6.11.9. Lenguaje Unificado Modelado (UML)

Booch (2008). UML ha sido creado por los expertos en la metodología orientada a objetos tales como Grady Booch, Ivar Jacobson, y James

Rumbaugh en Rational Software. Su objetivo era unificar los diversos sistemas que había y crear un lenguaje de modelado con las mejores características de cada uno.

La técnica central en el UML es el Modelamiento en Objetos que es un lenguaje que permite la especificación de clases, sus datos o atributos (privados) y métodos (públicos), herencia y otras relaciones entre las clases.

Los elementos notacionales que presenta el UML pretenden ser un lenguaje común para el modelamiento de cualquier sistema y se agrupan en tipos de diagrama:

1. Diagrama de Clases
2. Diagrama de Casos de Uso
3. Diagrama de Secuencia
4. Diagrama de Colaboración
5. Diagrama de Actividades
6. Diagrama conceptual

2.6.11.10. Diagrama de Clases

Booch (2008). Un **diagrama de clases** es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro. Ejemplo.

2.6.11.11. Diagrama de Caso de Uso

Booch (2008). Un diagrama de caso de uso describe lo que hace un sistema desde el punto de vista de un observador externo, debido a esto, un diagrama de este tipo generalmente es de los más sencillos de interpretar en UML, ya

que su razón de ser se concentra en un Que hace el sistema, a diferencia de otros diagramas UML que intentan dar respuesta a un Como logra su comportamiento el sistema.

2.6.11.12. Diagrama de Secuencia

Booch (2008). Los diagramas de secuencia muestran el intercambio de mensajes (es decir la forma en que se invocan) en un momento dado. Los diagramas de secuencia ponen especial énfasis en el orden y el momento en que se envían los mensajes a los objetos.

En los diagramas de secuencia, los objetos están representados por líneas intermitentes verticales, con el nombre del objeto en la parte más alta. El eje de tiempo también es vertical, incrementándose hacia abajo, de forma que los mensajes son enviados de un objeto a otro en forma de flechas con los nombres de la operación y los parámetros.

2.6.11.13. Diagrama de Colaboración.

Booch (2008). Los diagramas de colaboración muestran las interacciones que ocurren entre los objetos que participan en una situación determinada.

En los diagramas de colaboración los mensajes enviados de un objeto a otro se representan mediante flechas, mostrando el nombre del mensaje, los parámetros y la secuencia del mensaje. Los diagramas de colaboración están indicados para mostrar una situación o flujo programa específicos y son unos de los mejores tipos de diagramas para demostrar o explicar rápidamente un proceso dentro de la lógica del programa.

2.6.11.14. Diagrama de Actividades.

Booch (2008). Los diagramas de actividad describen la secuencia de las actividades en un sistema. Los diagramas de actividad son una forma especial de los diagramas de estado, que únicamente (o mayormente) contienen actividades.

2.6.11.15. Modelo Conceptual.

Booch (2008). Un modelo conceptual explica los conceptos significativos en un dominio del problema.

Es el artefacto más importante del análisis. La designación del modelo conceptual ofrece la ventaja de subrayar fuertemente una concentración en el dominio del problema más no las entidades del software.

2.6.11.16. Metodología Xtreme Programming

Agile (2009). Esta metodología promueve los siguientes valores:

- **Comunicación**

La Metodología Xtreme Programming se nutre del ancho de banda más grande que se puede obtener cuando existe algún tipo de comunicación: la comunicación directa entre personas. Es muy importante entender cuáles son las ventajas de este medio. Cuando dos (o más) personas se comunican directamente pueden no solo consumir las palabras formuladas por la otra persona, sino que también aprecian los gestos, miradas, etc. que hace su compañero. Sin embargo, en una conversación mediante el correo electrónico, hay muchos factores que hacen de esta una comunicación, por así decirlo, mucho menos efectiva.

- **Coraje**

El coraje es un valor muy importante dentro de la programación extrema. Un miembro de un equipo de desarrollo extremo debe de tener el coraje de exponer sus dudas, miedos, experiencias sin "embellecer" éstas de ninguna de las maneras. Esto es muy importante ya que un equipo de desarrollo extremo se basa en la confianza para con sus miembros. Faltar a esta confianza es una falta más que grave.

- **Simplicidad**

Dado que no se puede predecir cómo va a ser en el futuro, el software que se está desarrollando; un equipo de programación extrema intenta mantener el software lo más sencillo posible. Esto quiere decir que no se va a invertir ningún esfuerzo en hacer un desarrollo que en un futuro pueda llegar a tener valor. En el XP frases como "...en un futuro vamos a necesitar..." o "Haz un sistema genérico de..." no tienen ningún sentido ya que no aportan ningún valor en el momento.

- **Feedback**

Agile (2009). La agilidad se define (entre otras cosas) por la capacidad de respuesta ante los cambios que se van haciendo necesarios a lo largo del camino. Por este motivo uno de los valores que nos hace más ágiles es el continuo seguimiento o feedback que recibimos a la hora de desarrollar en un entorno ágil de desarrollo. Este feedback se toma del cliente, de los miembros del equipo, en cuestión de todo el entorno en el que se mueve un equipo de desarrollo ágil.

Para ello se fundamenta en las siguientes doce prácticas:

- 1. Planificación incremental**

Agile (2009). La Programación Extrema asume que la planificación nunca será

perfecta, y que variará en función de cómo varíen las necesidades del negocio. Por tanto, el valor real reside en obtener rápidamente un plan inicial, y contar con mecanismos de feedback que permitan conocer con precisión dónde estamos. Como es lógico, la planificación es iterativa: un representante del negocio decide al comienzo de cada iteración qué características concretas se van a implementar.

El objetivo de la XP es generar versiones de la aplicación tan pequeñas como sea posible, pero que proporcionen un valor adicional claro, desde el punto de vista del negocio. A estas versiones se las denomina releases.

Una reléase cuenta con un cierto número de historias. La historia es la unidad de funcionalidad en un proyecto XP, y corresponde a la mínima funcionalidad posible que tiene valor desde el punto de vista del negocio. Durante cada iteración se cierran varias historias, lo que hace que toda iteración añada un valor tangible para el cliente.

Es fundamental en toda esta planificación la presencia de un representante del cliente, que forma parte del equipo y que decide cuáles son las historias más valiosas. Estas historias son las que se desarrollarán en la iteración actual.

La obtención de feedback que permita llevar a cabo estimaciones precisas es fundamental. Se hacen estimaciones para cada historia, de modo que en cuanto se comienzan a tener datos históricos, éstos se utilizan para hacer que las siguientes estimaciones sean más precisas.

Como se puede ver, y como siempre ocurre con la Programación Extrema, el enfoque utilizado para llevar a cabo la planificación es eminentemente pragmático. Gran parte de la eficacia de este modelo de planificación deriva de una división clara de responsabilidades, que tiene en cuenta las necesidades del negocio en todo momento. Dentro de esta división, el representante del cliente tiene las siguientes responsabilidades:

- Decidir qué se implementa en cada reléase o iteración.

- Fijar las fechas de fin de la reléase, recortando unas características o añadiendo otras.
- Priorizar el orden de implementación, en función del valor de negocio.

Las responsabilidades del equipo de desarrollo son las siguientes:

- Estimar cuánto tiempo llevará una historia: este feedback es fundamental para el cliente, y puede llevarle a reconsiderar qué historias se deben incluir en una iteración.
- Proporcionar información sobre el coste de utilizar distintas opciones tecnológicas.
- Organizar el equipo.
- Estimar el riesgo de cada historia.
- Decidir el orden de desarrollo de historias dentro de la iteración.

2. Testing

Agile (2009). La ejecución automatizada de tests es un elemento clave de la XP. Existen tanto tests internos (o tests de unidad), para garantizar que el mismo es correcto, como tests de aceptación, para garantizar que el código hace lo que debe hacer. El cliente es el responsable de definir los tests de aceptación, no necesariamente de implementarlos. Él es la persona mejor cualificada para decidir cuál es la funcionalidad más valiosa.

El hecho de que los tests sean automatizados es el único modo de garantizar que todo funciona: desde el punto de vista de la XP, si no hay tests, las cosas sólo funcionan en apariencia. Aún más, si un test no está automatizado, no se le puede considerar como tal.

El objetivo de los tests no es corregir errores, sino prevenirlos. Por ejemplo, los tests siempre se escriben antes que el código a testear, no después: esto aporta un gran valor adicional, pues fuerza a los desarrolladores a pensar cómo se va a usar el código que escriben, poniéndolos en la posición de consumidores del software. Elaborar los tests exige pensar por adelantado

cuáles son los problemas más graves que se pueden presentar, y cuáles son los puntos dudosos. Esto evita muchos problemas y dudas, en lugar de dejar que aparezcan "sobre la marcha".

Un efecto lateral importante de los tests es que dan una gran seguridad a los desarrolladores: es posible llegar a hacer cambios más o menos importantes sin miedo a problemas inesperados, dado que proporcionan una red de seguridad. La existencia de tests hace el código muy maleable.

3. Programación en parejas

Agile (2009). La XP incluye, como una de sus prácticas estándar, la programación en parejas. Nadie programa en solitario, siempre hay dos personas delante del ordenador. Ésta es una de las características que más se cuestiona al comienzo de la adopción de la XP dentro de un equipo, pero en la práctica se acepta rápidamente y de forma entusiasta.

El principal argumento contra la programación en parejas es que es improductiva. Esta idea se basa en el hecho de que dos programadores "programan el doble por separado". Esto sería así si no fuera por las siguientes razones:

- a. El hecho de que todas las decisiones las tomen al menos dos personas proporciona un mecanismo de seguridad enormemente valioso.
- b. Con dos personas responsabilizándose del código en cada momento, es menos probable que se caiga en la tentación de dejar de escribir tests, etc., algo fundamental para mantener el código en buena forma. Es muy difícil que dos personas se salten tareas por descuido o negligencia
- c. El hecho de programar en parejas permite la dispersión de know-how por todo el equipo. Este efecto es difícil de conseguir de otro modo, y hace que la incorporación de nuevos miembros al equipo sea mucho más rápida y eficaz.
- d. El código siempre está siendo revisado por otra persona. La revisión de

código es el método más eficaz de conseguir código de calidad, algo corroborado por numerosos estudios, muchos de los cuáles son anteriores a la Programación Extrema.

- e. En contra de lo que pueda parecer, los dos desarrolladores no hacen lo mismo: mientras el que tiene el teclado adopta un rol más táctico, el otro adopta un rol más estratégico, preguntándose constantemente si lo que se está haciendo tiene sentido desde un punto de vista global.
- f. Los datos indican que la programación en parejas es realmente más eficiente. Si bien se sacrifica un poco de velocidad al comienzo, luego se obtiene una velocidad de crucero muy superior. Esto contrasta con lo que ocurre en la mayor parte de los proyectos, en los que se arranca con una velocidad enorme pero rápidamente se llega a un estado muy parecido a la parálisis, en el que progresos cada vez más pequeños consumen cantidades de tiempo cada vez más grandes. Todos conocemos proyectos que se pasan el 50% del tiempo en el estado de "finalizado al 90%".

4. Refactorización

Agile (2009). A la hora de la verdad, el código de la mayor parte de las aplicaciones empieza en un razonable buen estado, para luego deteriorarse de forma progresiva. El coste desorbitado del mantenimiento, modificación y ampliación de aplicaciones ya existente se debe en gran parte a este hecho.

Uno de los objetivos de la XP es mantener la curva de costes tan plana como sea posible, por lo que existen una serie de mecanismos destinados a mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez. A este proceso básico para mantener el código en buena forma se le llama refactorización.

La refactorización no sólo sirve para mantener el código legible y sencillo: también se utiliza cuando resulta conveniente modificar código existente para hacer más fácil implementar nueva funcionalidad.

5. Diseño simple

Agile (2009). Otra práctica fundamental de la Programación Extrema es utilizar diseños tan simples como sea posible. El principio es "utilizar el diseño más sencillo que consiga que todo funcione". Se evita diseñar características extra porque a la hora de la verdad la experiencia indica que raramente se puede anticipar qué necesidades se convertirán en reales y cuáles no. La XP nos pide que no vivamos bajo la ilusión de que un diseño puede resolver todas o gran parte de las situaciones futuras: lo que parece necesario cambia con frecuencia, es difícil acertar a priori.

Es obvio que, si no vamos a anticipar futuras necesidades, debemos poder modificar el diseño si alguna de estas se materializa. La XP soporta estas modificaciones gracias a los tests automatizados. Estos permiten hacer cambios importantes gracias a la red de protección que proporcionan. La refactorización, que hace que el código existente sea claro y sencillo, también ayuda a hacer factibles las modificaciones.

La XP define un "diseño tan simple como sea posible" como aquél que:

- Pasa todos los tests.
- No contiene código duplicado.
- Deja clara la intención de los programadores (enfatisa el qué, no el cómo) en cada línea de código.
- Contiene el menor número posible de clases y métodos.

6. Propiedad colectiva del código

Agile (2009). La XP aboga por la propiedad colectiva del código. En otras palabras, todo el mundo tiene autoridad para hacer cambios a cualquier código, y es responsable de ellos. Esto permite no tener que estar esperando a otros cuando todo lo que hace falta es algún pequeño cambio.

Por supuesto, cada cuál es responsable de las modificaciones que haga. El

principio básico es "tú lo rompes, tú lo arreglas, no importa si está en el código propio o en el de otros".

Por último, vale la pena tener en cuenta que la existencia de tests automatizados impide que se produzca un desarrollo anárquico, al ser cada persona responsable de que todos los tests se ejecuten con éxito al incorporar los cambios que ha introducido al programa.

7. Integración continua

Agile (2009). En muchos casos la integración de código produce efectos laterales imprevistos, y en ocasiones la integración puede llegar a ser realmente traumática, cuando dejan de funcionar cosas por motivos desconocidos. La Programación Extrema hace que la integración sea permanente, con lo que todos los problemas se manifiestan de forma inmediata, en lugar de durante una fase de integración más o menos remota.

La existencia de una fase de integración separada tiene dos efectos laterales indeseables: se empieza a hacer codificación "yo-yo", en la que todo el mundo modifica código "sólo para que funcione, ya lo ajustaremos", y hace que se acumulen defectos. Evitar que se acumulen defectos es muy importante para la XP, como lo es el conseguir que los defectos que cada programador inyecta los elimine él mismo.

8. Cliente en el equipo

Agile (2009). Algunos de los problemas más graves en el desarrollo son los que se originan cuando el equipo de desarrollo toma decisiones de negocio críticas. Esto no debería ocurrir, pero a la hora de la verdad con frecuencia no se obtiene feedback del cliente con la fluidez necesaria: el resultado es que se ha de optar por detener el avance de los proyectos, o porque desarrollo tome una decisión de negocio. Por otra parte, los representantes del negocio también suelen encontrarse con problemas inesperados debido a que tampoco

reciben el feedback adecuado por parte de los desarrolladores.

La XP intenta resolver este tipo de problemas integrando un representante del negocio dentro del equipo de desarrollo. Ésta persona siempre está disponible para resolver dudas y para decidir qué y qué no se hace en cada momento, en función de los intereses del negocio. Debido a su inmersión dentro del equipo, y a que es él quien decide qué y qué no se hace, junto con los tests que verifican si la funcionalidad es la correcta y deseada, esta persona obtiene un feedback absolutamente realista del estado del proyecto.

9. Releaseas pequeñas

Agile (2009). Siguiendo la política de la XP de dar el máximo valor posible en cada momento, se intenta liberar nuevas versiones de las aplicaciones con frecuencia. Éstas deben ser tan pequeñas como sea posible, aunque deben añadir suficiente valor como para que resulten valiosas para el cliente.

10. Semanas de 40 horas

Agile (2009). La Programación Extrema lleva a un modo de trabajo en que el equipo está siempre al 100%. Una semana de 40 horas en las que se dedica la mayor parte del tiempo a tareas que suponen un avance puede dar mucho de sí, y hace innecesario recurrir a sobreesfuerzos -excepto en casos extremos.

Además, el sobreesfuerzo continuado pronto lleva a un rendimiento menor y a un deterioro de la moral de todo el equipo.

11. Estándares de codificación

Agile (2009). Para conseguir que el código se encuentre en buen estado y que cualquier persona del equipo pueda modificar cualquier parte del código es imprescindible que el estilo de codificación sea consistente. Un estándar de

codificación es necesario para soportar otras prácticas de la XP.

Sin embargo, la XP también es pragmática en esto, y apuesta por establecer un número mínimo de reglas: el resto se irán pactando de-facto. Esto evita un ejercicio inicial más o menos estéril.

12. Uso de Metáforas

Agile (2009). La comunicación fluida es uno de los valores más importantes de la Programación Extrema: la programación en parejas, el hecho de incorporar al equipo una persona que represente los intereses del negocio y otras prácticas son valiosas entre otras cosas porque potencian enormemente la comunicación.

Para conseguir que la comunicación sea fluida es imprescindible, entre otras cosas, utilizar el vocabulario del negocio. También es fundamental huir de definiciones abstractas. Dicho de otro modo, la XP no pretende seguir la letra de la ley, sino su espíritu. Dentro de este enfoque es fundamental buscar continuamente metáforas que comuniquen intenciones y resulten descriptivas, enfatizando el qué por delante del cómo.

La metodología XP es una metodología ágil

- Los individuos e interacciones son más importantes que los procesos y herramientas.
Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- Software que funcione es más importante que documentación exhaustiva.

Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.

- La colaboración con el cliente es más importante que la negociación de contratos.

La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.

- La respuesta ante el cambio es más importante que el seguimiento de un plan.

Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

2.6.11.17. Metodología OMT

2.6.11.18. Introducción.

Rumbaugh (2010). La metodología OMT (Object Modeling Technique) fue creada por James Rumbaugh y Michael Blaha en 1991, mientras James dirigía un equipo de investigación de los laboratorios General Electric.

OMT es una de las metodologías de análisis y diseño orientados a objetos, más maduros y eficientes que existen en la actualidad. La gran virtud que aporta esta metodología es su carácter de abierta (no propietaria), que le permite ser de dominio público y en consecuencia, sobrevivir con enorme vitalidad. Esto facilita su evolución para acoplarse a todas las necesidades actuales y futuras de la ingeniería de software.

Las fases que conforman a la metodología OMT son:

- **Análisis.** El analista construye un modelo del dominio del problema, mostrando sus propiedades más importantes. El modelo de análisis es una abstracción resumida y precisa de lo que debe de hacer el sistema deseado y no de la forma en que se hará. Los elementos del modelo deben ser conceptos del dominio de aplicación y no conceptos informáticos tales como estructuras de datos. Un buen modelo debe poder ser entendido y criticado por expertos en el dominio del problema que no tengan conocimientos informáticos.
- **Diseño del sistema.** El diseñador del sistema toma decisiones de alto nivel sobre la arquitectura del mismo. Durante esta fase el sistema se organiza en subsistemas basándose tanto en la estructura del análisis como en la arquitectura propuesta. Se selecciona una estrategia para afrontar el problema.
- **Diseño de objetos.** El diseñador de objetos construye un modelo de diseño basándose en el modelo de análisis, pero incorporando detalles de implementación. El diseño de objetos se centra en las estructuras de datos y algoritmos que son necesarios para implementar cada clase. OMT describe la forma en que el diseño puede ser implementado en distintos lenguajes (orientados y no orientados a objetos, bases de datos, etc.).
- **Implementación.** Las clases de objetos y relaciones desarrolladas durante el análisis de objetos se traducen finalmente a una implementación concreta. Durante la fase de implementación es importante tener en cuenta los principios de la ingeniería del software de forma que la correspondencia con el diseño sea directa y el sistema implementado sea flexible y extensible. No tiene sentido que utilicemos AOO y DOO de forma que potenciemos la reutilización de código y la correspondencia entre el dominio del problema y el sistema informático, si luego perdemos todas estas ventajas con una implementación de mala calidad.

Rumbaugh (2010). La metodología OMT emplea tres clases de modelos para describir el sistema:

- **Modelo de objetos.** Describe la estructura estática de los objetos del sistema (identidad, relaciones con otros objetos, atributos y operaciones). El modelo de objetos proporciona el entorno esencial en el cual se pueden situar el modelo dinámico y el modelo funcional. El objetivo es capturar aquellos conceptos del mundo real que sean importantes para la aplicación. Se representa mediante diagramas de objetos.
- **Modelo dinámico.** Describe los aspectos de un sistema que tratan de la temporización y secuencia de operaciones (sucesos que marcan los cambios, secuencias de sucesos, estados que definen el contexto para los sucesos) y la organización de sucesos y estados. Captura el control, aquel aspecto de un sistema que describe las secuencias de operaciones que se producen sin tener en cuenta lo que hagan las operaciones, aquello a lo que afectan o la forma en que están implementadas. Se representa gráficamente mediante diagramas de estado.
- **Modelo funcional.** Describe las transformaciones de valores de datos (funciones, correspondencias, restricciones y dependencias funcionales) que ocurren dentro del sistema. Captura lo que hace el sistema, independientemente de cuando se haga o de la forma en que se haga. Se representa mediante diagramas de flujo de datos

2.6.11.19. Modelo de Objetos

Rumbaugh (2010). Esta es la parte principal de la Técnica para modelado ya que se fundamenta en la teoría de OO. La definición clara de las entidades que intervienen en el sistema es un paso inicial necesario para poder definir qué transformaciones ocurren en ellas y cuándo se producen estas transformaciones. Esta forma de pensar es inherente al paradigma de OO donde las clases y su jerarquía determinan el sistema.

Los diagramas de objetos permiten representar gráficamente los objetos, las

clases y sus relaciones mediante dos tipos de diagramas: los diagramas de clases y los diagramas de casos concretos (instancias). Los diagramas de clases describen las clases que componen el sistema y que permitirán la creación de casos concretos, los diagramas de casos concretos describen la manera en que los objetos del sistema se relacionan y los casos concretos que existen en el sistema de cada clase. En los diagramas que componen este modelo se pueden representar los siguientes elementos del sistema: objetos y clases, atributos, operaciones, y relaciones o asociaciones.

2.6.11.20. Clases y Objetos

Rumbaugh (2010). Los objetos y sus componentes se representan gráficamente en OMT de forma que es posible obtener una idea de los elementos que intervienen en el sistema estudiando el modelo. Los elementos y sus características con representación gráfica son los siguientes:

- **Objetos.** Un objeto es, sencillamente, algo que tiene sentido en el contexto de la aplicación. Se definirá un objeto como un concepto, abstracción o cosa con límites bien definidos y con significado a efectos del problema que se tenga entre manos.
- **Clases.** Describe un grupo de objetos con propiedades (atributos) similares, con relaciones comunes con otros y con una semántica común.
- **Diagramas de objetos.** Proporcionan un anotación gráfica formal para el modelado de objetos, clases y sus relaciones entre sí, son útiles, tanto para el modelado abstracto como, para diseñar programas reales. Hay dos tipos de diagramas de objetos
- **Diagrama de clases.** Esquema, patrón o plantilla para describir muchas instancias de datos posibles.
- **Diagrama de instancias.** Describe la forma en que un cierto conjunto de objetos se relacionan entre sí.
- **Atributos.** Los objetos pertenecientes a una clase presentan características que en OMT se denominan atributos. Sin embargo, no se deben confundir los atributos, que son características que todos los objetos de una

clase comparten, con otros objetos que pueden formar parte del objeto que estamos tratando.

- Operaciones y métodos. Del mismo modo que los objetos en OMT se pueden representar las operaciones que se realizan sobre ellos o que éstos realizan sobre otros objetos del sistema. Los objetos realizan acciones sobre otros objetos y definen acciones que se realizan sobre ellos mismos. Los objetos de una misma clase comparten estas operaciones, aunque también pueden añadir otras nuevas que no se definan en su clase a medida que se especializa el objeto en otras subclases. También pueden redefinir las operaciones en estas especializaciones ignorando las definiciones realizadas en las superclases. Las operaciones pueden llevar implícito el objeto sobre el que se realizan o que realiza la acción, de forma que es posible tener una misma operación que se efectúe de manera distinta según el objeto sobre el que se aplique. La implementación de las operaciones para cada uno de los objetos diferentes (o subclases) se denomina método. Los métodos implementan en cada una de las clases de forma específica para los objetos que representa.

2.6.11.21. Enlaces y Asociaciones

Rumbaugh (2010). Las relaciones entre clases determinan el comportamiento del sistema y constituyen una parte muy importante del mismo ya que mediante las relaciones definimos la forma en que los objetos se comunican, lo que también se conoce como comportamiento.

Además las relaciones tienen una serie de características que son de interés para el modelado del sistema.

- Relaciones. En OMT se identifican a través de enlaces: conexiones físicas o conceptuales entre casos concretos de objetos. Una asociación en OMT abstrae un conjunto de enlaces con una estructura y un significado comunes. Desde el punto de vista de la implementación, una asociación es un puntero que apunta desde un objeto a otro.

- Multiplicidad. Este término se encuentra relacionado con las asociaciones e indica el número de casos concretos de una clase que puede relacionarse con otro caso concreto. Las relaciones más frecuentes son las binarias, aunque pueden darse de cualquier orden.

2.6.11.22. Conceptos Avanzados de Enlaces y Asociaciones

Rumbaugh (2010). Atributos de los enlaces. Los enlaces así como los objetos pueden tener atributos, que son propiedades de los enlaces de una asociación.

- Modelado de una asociación en forma de clase. A veces resulta conveniente modelar las asociaciones como clases en lugar de como relaciones, cuando los enlaces pueden participar en asociaciones con otros objetos o están sometidos a operaciones.
- Clasificación. También podemos encontrar en una asociación de objetos que pertenecen a una clase con multiplicidad "muchos" que deben estar ordenados. Esta característica de los objetos es una restricción, ya que implica una condición que deben cumplir los elementos de la clase.
- Nombre de rol. Las asociaciones conectan clases u objetos que pertenecen a dichas clases, pero en ocasiones necesitamos restringir el conjunto de los objetos que se relacionan dentro de la clase. Mediante estas restricciones podemos especificar qué objetos se relacionan. Podemos conseguir este objetivo mediante los llamados nombres de rol. Un nombre de rol es un atributo derivado de una clase cuyo valor es un conjunto de objetos relacionados. Los nombres de rol se utilizan cuando las asociaciones se producen entre objetos de la misma clase ya que suelen producirse entre subconjuntos de esas clases. Una asociación binaria puede tener dos roles, uno por cada extremo de la asociación que son identificados por sus nombres.
- Cualificación. Las asociaciones muchos a muchos y uno a muchos pueden ser calificadas mediante un elemento calificador que reduce el conjunto de objetos relacionados, indicando un subconjunto de la clase que se califica en la asociación. Las asociaciones que se pueden calificar son las de

multiplicidad uno a muchos y muchos a muchos.

- Agregación. Las relaciones de agregación son para OMT formas de asociación del tipo "es parte de", como tales se definen entre una clase agregado y una clase componente y se indican con un rombo en la parte de la clase que actúa como continente. Las relaciones de agregación se establecen en los llamados objetos compuestos que contienen otros objetos y éstos pueden ser de dos tipos: aquellos que tienen existencia física más allá del objeto agregado, y los que no pueden existir sin el objeto agregado.

2.6.11.23. Generalización y Herencia

Rumbaugh (2010). En el paradigma de la orientación a objetos uno de los elementos más importantes es la herencia. La cualidad que permite que los objetos hereden las características (atributos) y las operaciones (métodos) dentro de una estructura jerárquica conlleva una serie de consecuencias de máxima relevancia a la hora de diseñar un sistema informático. Los objetos heredan un comportamiento que puede ser modificado y unas estructuras de datos de forma que se permite y se facilita la reutilización de las clases y del código que implementa sus funcionalidades.

Ambos conceptos van unidos: herencia y estructura jerárquica, de forma que la herencia se produce por la existencia de una estructura entre los componentes del sistema y la estructura se consigue en la implementación del código a través de la herencia en los lenguajes OO.

La herencia está íntimamente relacionada con la forma concreta en que un lenguaje implementa la generalización, que es un término más abstracto.

- La generalización es la relación que existe entre una clase y las subclases que se derivan de la misma. A partir de una versión "en bruto" se generan versiones más especializadas de la misma que añaden características y operaciones. La superclase es la versión general y las subclases son

especializaciones de la superclase. Las generalizaciones pueden tener discriminadores que indican qué aspecto de la superclase está siendo utilizado para obtener subclases más concretas.

- Anulación. Al implementar la herencia nos encontramos en numerosas ocasiones que las subclases redefinen operaciones que ya han sido definidas en las superclases. Las razones para esta nueva implementación de operaciones que existen en las superclases son variadas, a veces simplemente se le añaden nuevas acciones que van en consonancia con las nuevas características que añade la subclase; otras, se consigue optimizar las operaciones debido a que las subclases tienen características nuevas que lo permiten, y a veces se produce por una mala práctica de análisis donde no se prevén las operaciones de manera óptima.

Se han propuesto una serie de reglas a la hora de implementar la herencia para minimizar los errores y maximizar la reutilización de código:

- a. Las operaciones de consulta, aquellas que no modifican valores de atributos, se heredan por todas las subclases.
- b. Las operaciones de actualización, que modifican valores de atributos, se heredan por todas las subclases y se añaden las nuevas operaciones para aquellas que añadan atributos.
- c. Las operaciones de actualización que se realizan sobre atributos que tengan algún tipo de restricción o asociación, se bloquean para nuevas subclases.
- d. Las operaciones no pueden volver a definirse para hacer que se comporten de distinta manera de cara al exterior, es decir; todos los métodos concretos de una operación deben tener el mismo protocolo.
- e. Las operaciones se pueden refinar añadiendo comportamientos en las subclases.

2.6.11.24. Agrupación de entidades

Rumbaugh (2010). Los elementos que hemos estudiado en el Modelo de Objetos se pueden agrupar para construir el modelo completo, así, las clases, las asociaciones y las generalizaciones forman lo que se denomina módulo y varios módulos forman el modelo de objetos. En un módulo no se deben repetir los nombres de las clases y de las asociaciones, aunque se puede hacer referencia a la misma clase dentro de distintos módulos. También se definen las denominadas hojas que se utilizan para descomponer un Modelo de Objetos en unidades que podemos manejar. Una hoja es una parte de un módulo que podemos manejar con facilidad, sea en el formato que sea.

2.6.11.25. Modelo Avanzado de Datos

Rumbaugh (2010).

- Clases abstractas. En ocasiones puede ser de utilidad tener clases que definan propiedades y operaciones de forma general, dejando para sucesivos refinamientos la implementación concreta de las mismas. Una clase abstracta es precisamente una clase donde se introducen métodos y datos que se definirán en las subclases de la misma. Las clases abstractas siempre tienen que tener clases derivadas donde se especificarán las operaciones que no se hayan definido en la clase abstracta, nunca podrán tener objetos, se utilizan como clases bases o superclases de otras clases. La existencia de este tipo de clases facilita aún más la posibilidad de abstracción del modelo, dejando para posteriores refinamientos del problema la definición más exhaustiva de operaciones y datos. La implementación concreta de las clases abstractas depende del lenguaje OO. En OMT una clase abstracta se identifica cuando no tiene casos concretos (instancias) pero una subclase suya sí los tiene. Por ejemplo, una clase abstracta números podría definir una operación multiplicar que no se implementara hasta la definición de las subclases números enteros, números reales y números complejos, teniendo la información precisa en

cada una de las subclases de la forma en que se multiplican los números.

- Herencia múltiple. La herencia múltiple es una característica que algunos sistemas OO poseen, mediante la cual es posible que una clase herede de varias superclases al mismo tiempo. Sin embargo, la herencia múltiple aumenta radicalmente la complejidad de los sistemas que la implementan ya que la búsqueda de las operaciones se dificulta cuando no se define en la clase derivada y hay que realizarla en las superclases.
- Clave candidata. No es más que un conjunto mínimo de atributos que define de forma única un objeto o enlace. Es decir, mediante una clave candidata tenemos definido el objeto o el enlace con una serie de atributos de forma que se distingue del resto de objetos o enlaces. Las claves candidatas son restricciones y, por tanto, se representan como tales en OMT. Para encontrar una clave candidata en una asociación donde intervienen más de dos clases, debemos definir qué combinaciones de clases o enlaces en la asociación de los elementos definen la tercera clase o enlace de forma única.
- Restricciones. El modelo de objetos contiene diferentes entidades como son los objetos, las clases, los atributos, los enlaces y las asociaciones. Cada una de estas entidades tiene una serie de características inherentes a su naturaleza dentro del sistema que estamos modelando. Las características definen su significado dentro del sistema y también sus limitaciones. Estas limitaciones se denominan en el Análisis restricciones. Las restricciones pueden ser muy complejas y en este caso no pueden estudiarse en el modelo de objetos, sino que se especificarán en el modelo funcional. Las que intervienen en el modelo de objetos se representan mediante llaves junto a la entidad a que se refieran. Cuando la restricción implica más de una clase, se indica mediante una flecha discontinua que une los elementos que se vean implicados en la restricción. Las asociaciones también pueden tener restricciones.

2.6.11.26. Construcción de un modelo de objetos

Rumbaugh (2010).

- Identificar las clases de objetos.
- Iniciar un diccionario de datos que contenga descripciones de clases, atributos y asociaciones.
- Agregar asociaciones entre clases.
- Agregar atributos a objetos y ligas.
- Organizar y simplificar las clases de objetos usando herencia.
- Probar las rutas de acceso usando escenarios e iterar los pasos anteriores según sea necesario.
- Agrupar las clases en módulos, basándose en "acoplamiento cercano" y función relacionada.

2.6.11.27. Modelo Dinámico

Rumbaugh (2010). Los aspectos del sistema que están relacionados con el tiempo y con los cambios constituyen el modelo dinámico.

Los conceptos más importantes del modelado dinámico son los sucesos, que representan estímulos externos, y los estados, que representan los valores de los objetos. El diagrama de estados va a representar los sucesos y los estados que se dan en el sistema.

El modelo de objetos describe las posibles tramas de objetos, atributos y enlaces que pueden existir en un sistema. Los valores de los atributos y de los enlaces mantenidos por un objeto son lo que se denomina su estado. A lo largo del tiempo, los objetos se estimulan unos a otros, dando lugar a una serie de cambios en sus estados. Un estímulo individual proveniente de un objeto y que llega a otro es un suceso. La respuesta a un suceso depende del estado del objeto que lo recibe, y puede incluir un cambio de estado o el envío de otro suceso al remitente o a un tercer objeto. La trama de sucesos, estados y

transiciones de estados para una clase dada se puede abstraer y representar en forma de un diagrama de estados. El modelo dinámico consta de múltiples diagramas de estados, con un diagrama de estados para cada clase que posea un comportamiento dinámico importante, y muestra la trama de actividad para todo el sistema.

2.6.11.28. Sucesos y Estados

Rumbaugh (2010).

- Suceso. Un suceso, o evento, es algo que transcurre durante un período de tiempo. Un suceso puede preceder o seguir lógicamente a otro, o bien los dos sucesos pueden no estar relacionados. Se dice que dos sucesos que no tienen relación causal son concurrentes; no tienen efecto entre sí. Un suceso es una transmisión de información de dirección única entre un objeto y otro. No es como una llamada a subrutina, que proporciona un valor. En el modelo dinámico existe el concepto de clases de sucesos, que consiste en una estructura jerárquica, similar a la estructura de clases. Todo suceso aporta información de un objeto a otro. Los valores de datos aportados por un suceso son sus atributos.
- Escenarios y seguimiento de sucesos. Un escenario es una secuencia de sucesos que se produce durante una ejecución concreta de un sistema. El ámbito de un escenario es variable; puede incluir todos los sucesos del sistema, o que sean generados por ellos. Todo suceso transmite información de un objeto a otro. El primer paso para la construcción de escenario es identificar los objetos emisores y receptores de cada suceso. La secuencia de sucesos y los objetos que intercambian sucesos se pueden mostrar ambos en un escenario mejorado que se denomina de seguimiento traza de sucesos. El tiempo aumenta desde arriba hacia abajo, aunque en el seguimiento de sucesos no importa la temporización exacta, sino que importa la secuencia de los procesos.
- Estados. Un estado es una abstracción de los valores de los atributos y de los enlaces de un objeto. Los conjuntos de valores se agrupan dentro del

estado de acuerdo con aquellas propiedades que afecten al comportamiento del objeto. Un estado especifica la respuesta del objeto a los sucesos entrantes. La respuesta a un suceso recibido por un objeto puede variar cuantitativamente, dependiendo de los valores exactos de sus atributos, pero cualitativamente la respuesta es la misma para todos los valores dentro del mismo estado, y puede ser distinta para valores de distintos estados. La respuesta de un objeto a un suceso puede incluir una acción o un cambio de estado por parte del objeto. Un estado corresponde al intervalo entre dos sucesos recibidos por un objeto. Los sucesos representan puntos temporales; los estados representan intervalos de tiempo. Los estados suelen estar asociados con que el valor de un objeto satisfaga alguna condición. Al definir estados, ignoramos aquellos atributos que no afectan al comportamiento del objeto, y agrupamos en un único estado todas las combinaciones de valores de atributos y de enlaces que tienen una misma respuesta a los sucesos. Por supuesto, todo atributo tiene algún efecto sobre el comportamiento, o no tendría sentido, pero es frecuente que algunos atributos no afecten a la trama de control, y que se pueda pensar en ellos como valores de parámetros dentro de un estado. Tanto los sucesos como los estados dependen del nivel de abstracción utilizado. Los estados se pueden caracterizar de diferentes maneras, pero normalmente cada estado tiene un nombre y una descripción en la que se indica en la situación en la que se encuentra el sistema.

2.6.11.29. Diagrama de Estados

Rumbaugh (2010). Un diagrama de estados relaciona sucesos y estados. Cuando se recibe un suceso, el estado siguiente depende del actual, así como del suceso; un cambio de estado causado por un suceso es lo que se llama una transición. Un diagrama de estados es un grafo cuyos nodos son estados, y cuyos arcos dirigidos son transiciones rotuladas con nombres de sucesos.

El diagrama de estados especifica la secuencia de estados que causa una

cierta secuencia de sucesos. Si un objeto se encuentra en un cierto estado y se produce un suceso cuyo nombre corresponda al de una de sus transiciones, entonces el objeto pasa al estado que se encuentra en el extremo de destino de la transición. Se dice que la transición se dispara. Si hay más de una transición que sale de un estado, entonces el primer suceso que se produzca dará lugar a que se dispare la transición correspondiente.

Si se produce un suceso que no tiene ninguna transición que salga del estado actual, entonces el suceso se ignora. Una secuencia de sucesos se corresponde con un camino a través del grafo.

Un diagrama de estados describe el comportamiento de una sola clase de objetos. Dado que todas las instancias de una clase tienen el mismo comportamiento, todas ellas comparten el mismo diagrama de estados, por cuanto todas ellas comparten las mismas características de clase. Pero dado que todo objeto posee sus propios valores de atributos, cada objeto posee su propio estado, que es el resultado de la especial secuencia de sucesos que haya recibido. Todo objeto es independiente de los demás objetos, y procede a su paso.

Los diagramas de estados pueden representar ciclos vitales únicos o bien bucles continuos. Los diagramas de un solo uso representan objetos de duración finita y tienen estados iniciales y finales. Se entra en el estado inicial al crear el objeto; al entrar en el estado final estamos implicando la destrucción del objeto. Los diagramas de un solo uso son una "subrutina" del diagrama de estados, a la cual se puede hacer alusión desde distintos lugares en un diagrama de alto nivel.

El modelo dinámico es una colección de diagramas de estados que interactúan entre sí a través de sucesos compartidos.

2.6.11.30. Condiciones

Rumbaugh (2010). Una condición es una función Booleana lógica que tiene a objetos como valores. Las condiciones se pueden utilizar como protecciones en las transiciones. Una transición con protección se dispara cuando se produce su suceso, pero sólo si la condición de protección es verdadera.

2.6.11.31. Operaciones

Rumbaugh (2010). Los diagramas de estados tendrían muy poca utilidad si solamente describiesen tramas de sucesos. Una descripción de un objeto debe especificar lo que hace el objeto como respuesta a los sucesos.

Una actividad es una operación cuya realización requiere un cierto tiempo. Toda actividad está asociada a un estado. Entre las actividades se cuentan las operaciones continuas, tales como mostrar una imagen en una pantalla, así como las operaciones secuenciales que terminan por sí mismas después de un cierto intervalo de tiempo. Un estado puede controlar una actividad continua o una actividad secuencial que va avanzando hasta que termina o hasta que se produce un suceso que la hace finalizar prematuramente. La anotación "hacer: X" indica que la actividad secuencial X comienza al entrar en ese estado, y se detiene cuando ha finalizado. Si un suceso da lugar a una transición que sale de ese estado antes de que haya finalizado la actividad, entonces, la actividad finaliza de forma prematura.

Una acción es una operación instantánea que va asociada a un suceso. Una acción representa a una operación cuya duración es insignificante en comparación con la resolución del diagrama de estados. Realmente, no hay operaciones que sean instantáneas, pero se modelan de esta manera aquellas operaciones de las que no nos importa su estructura interna.

Las acciones también pueden representar operaciones internas de control, tales como dar valores a atributos o generar otros sucesos. Estas acciones no

tienen contrapartidas en el mundo real, sino que son mecanismos para estructurar el control dentro de una implementación.

2.6.11.32. Diagramas de Estados Anidados

Rumbaugh (2010). Los diagramas de estados se pueden estructurar, para hacer posibles unas descripciones concisas de sistemas complejos. Las formas de estructurar máquinas de estados son similares a las formas de estructurar los objetos: la generalización y la agregación. La generalización es el equivalente a expandir las actividades anidadas. Permite describir una actividad empleando un nivel alto, y expandirla después en un nivel más bajo añadiendo detalles, de forma similar a las llamadas a procedimientos anidados. Además, la generalización permite que los estados y los sucesos se dispongan en jerarquías de generalización con herencia de estructuras y comportamientos comunes, de forma similar a la herencia de atributos y de operaciones en las clases. La agregación permite que el estado se descomponga en componentes ortogonales, con una interacción limitada entre ellos, de forma similar a una jerarquía de agregación de objetos. La agregación es el equivalente a la concurrencia de estados. Los estados concurrentes suelen corresponderse con agregaciones de objetos, posiblemente de todo un sistema, que tengan partes que interactúen.

- Anidamiento de diagramas de estados. Una actividad de un estado se puede expandir en forma de diagrama de estados de nivel inferior, en el cual cada uno representará un paso de la actividad. Las actividades anidadas son diagramas de estados de un solo uso, con transiciones de entrada y de salida, parecidas a subrutinas.
- Generalización de estados. Un diagrama de estados anidados es en realidad una forma de generalización de estados. Un objeto que se encuentre en un estado del diagrama de alto nivel tiene que estar precisamente en uno de los estados del diagrama anidado. Los estados del diagrama anidado son todos ellos refinamientos del estado del diagrama de

alto nivel. Los estados pueden poseer subestados que hereden las transiciones de sus súperestados, del mismo modo que las clases poseen subclases que heredan los atributos y operaciones de sus superclases. Toda transición o acción que sea aplicable a un estado es aplicable también a todos sus subestados, a no ser que sea invalidada por una transición equivalente del subestado. Las transiciones de un súperestado son heredadas por todos sus subestados.

2.6.11.33. Concurrencia

Rumbaugh (2010).

- Concurrencia de Agregación. Un modelo dinámico describe un conjunto de objetos concurrentes, cada cual con su propio estado y con su propio diagrama de estados. Los objetos de todo sistema son inherentemente concurrentes, y pueden cambiar de estado independientemente. El estado de todo el sistema no se puede representar mediante un solo estado de un único objeto; es el producto de los estados de todos los objetos que lo componen. En muchos sistemas además, el número de objetos puede cambiar dinámicamente. Un diagrama de estados para un subsistema es una colección de los mismos, uno por cada componente. La agregación implica concurrencia. El estado agregado corresponde a los estados combinados de todos los diagramas que componen el subsistema. El estado agregado es un estado del primer diagrama y de todos los demás diagramas. En algunos casos los estados componentes interactúan. Las transiciones protegidas para un objeto pueden depender de que otro objeto se encuentre en un cierto estado. Esto permite la interacción entre diagramas de estados, manteniendo al mismo tiempo la modularidad.
- Concurrencia dentro de un objeto. La concurrencia dentro del estado de un único objeto surge cuando se puede descomponer el objeto en subconjuntos de atributos o de enlaces, cada uno de los cuales posee su propio subdiagrama.

2.6.11.34. Desarrollo de un modelo dinámico

Rumbaugh (2010).

- Preparar escenarios para las secuencias de interacción típicas.
- Identificar eventos entre objetos y preparar trazos de eventos para cada escenario.
- Preparar un diagrama de flujo de eventos para el sistema.
- Desarrollar un diagrama de estados para cada clase que tenga un comportamiento dinámico importante.
- Verificar que los eventos compartidos entre diagramas de estado sean consistentes y correctos.

2.6.11.35. Modelo Funcional

Rumbaugh (2010). El modelo funcional describe los cálculos existentes dentro del sistema siendo la tercera parte del modelado. Dentro del modelado del sistema, el modelo funcional especifica lo que sucede, el modelo dinámico cuándo sucede, y el modelo de objetos especifica a qué le sucede.

El modelo funcional muestra la forma en que se derivan los valores producidos en un cálculo a partir de los valores introducidos, sin tener en cuenta el orden en el cual se calculan los valores. Consta de múltiples diagramas de flujo de datos, que muestran el flujo de valores desde las entradas externas, a través de las operaciones y almacenes internos de datos hasta las salidas externas. También incluyen restricciones entre valores dentro del modelo de objetos. Los diagramas de flujo de datos no muestran el control ni tampoco información acerca de la estructura de los objetos; todo esto pertenece a los modelos dinámico y de objetos.

2.6.11.36. Diagramas de flujo de datos

Rumbaugh (2010). El modelo funcional consta de múltiples diagramas de flujo

de datos, que especifican el significado de las operaciones y de las restricciones. Un diagrama de flujo de datos (DFD) muestra las relaciones funcionales entre los valores calculados por un sistema, incluyendo los valores introducidos, los obtenidos, y los almacenes internos de datos. Un diagrama de flujo de datos es un grafo que muestra el flujo de valores de datos desde sus fuentes en los objetos mediante procesos que los transforman, hasta sus destinos en otros objetos. Un diagrama de flujo de datos no muestra información de control como puede ser el momento en que se ejecutan los procesos o se toman decisiones entre vías de datos alternativas; esta información pertenece al modelo dinámico. Un diagrama de flujo de datos no muestra la organización de los valores en objetos; esta información pertenece al modelo de objetos.

Un diagrama de flujo de datos contiene procesos que transforman datos, flujos de datos que los trasladan, objetos actores que producen y consumen datos, y de almacenes de datos que los almacenan de forma pasiva.

- **Procesos.** Un proceso transforma valores de datos. Los procesos del más bajo nivel son funciones puras, sin efectos laterales. Un grafo completo de flujo de datos es un proceso de alto nivel. Los procesos pueden tener efectos laterales si contienen componentes no funcionales, tales como almacenes de datos u objetos externos. El modelo funcional no especifica de forma única el resultado de un proceso que tenga efectos laterales, solamente indica las posibles vías funcionales; no muestra la que realmente se recorrerá. Los resultados de estos procesos dependen del comportamiento del sistema, según se especifica en el modelo dinámico. El diagrama muestra solamente el patrón de entradas y salidas. El cálculo de valores de salida a partir de los de entrada también debe ser especificado. Un proceso de alto nivel se puede expandir en todo un diagrama de flujo de datos, de forma muy parecida a la manera en que se puede expandir una subrutina en otra subrutina de nivel inferior. Eventualmente, la recursividad finaliza, y los procesos atómicos deben describirse directamente, en lenguaje natural, mediante ecuaciones matemáticas, o por algún otro medio.

Los procesos se implementan como métodos de operaciones que se aplican a clases de objetos. El objeto destino suele ser uno de los flujos de entrada, sobre todo si esa misma clase de objeto es también un flujo de salida.

- Flujo de datos. Un flujo de datos conecta la salida de un objeto o proceso con la entrada de otro objeto o proceso. Representa un valor de datos intermedio dentro de un cálculo que no es modificado por el flujo de datos. El mismo valor se puede enviar a varios lugares. En algunas ocasiones un valor de datos agregado se descompone en sus componentes, cada uno de los cuales va a un proceso diferente. La combinación de varios componentes en un valor agregado es justamente lo contrario. Cada flujo de datos representa un valor en algún momento del cálculo. Los flujos de datos internos al diagrama representan valores intermedios dentro de un cálculo, y o tienen necesariamente ningún significado en el mundo real. Los flujos en la frontera de un diagrama de flujo de datos son sus entradas y salidas. Estos pueden ser inconexos (si el diagrama es un fragmento de un sistema completo), o bien pueden estar conectados con objetos.

2.6.11.37. Especificación de Operaciones

Rumbaugh (2010). Los procesos de los diagramas de flujo deben ser implementados eventualmente como operaciones que se aplican a objetos. Todo proceso atómico del más bajo nivel es una operación. Los procesos de nivel superior también se pueden considerar operaciones, aun cuando una implementación pueda estar organizada de forma distinta del diagrama de datos que representa como consecuencia de la optimización. Toda operación se podrá especificar de diferentes maneras, entre las que están:

- Funciones matemáticas,
- Tablas de valores de entrada y salida,
- Ecuaciones que especifican la salida dependiendo de la entrada,

condiciones previas y posteriores,

- Tablas de decisión,
- Pseudocódigo,
- Lenguaje natural.

La especificación de una operación se compone de dos partes. La primera de ellas es la que indica la interfaz de la operación: los argumentos que requiere y los valores que proporciona. La segunda, es la que explica la transformación de los valores de entrada para producir los valores de salida.

La especificación externa de una operación describe solamente cambios visibles fuera de ella. Durante la implementación de una operación, se pueden crear valores internos por conveniencia o por optimización. Algunos pueden incluso formar parte del estado interno de un objeto. El propósito de la especificación es indicar lo que debe hacer una operación lógicamente, y no como debe ser implementada. Por tanto, el estado del objeto en sí debe dividirse en información visible externamente e información privada, interna. Los cambios del estado interno de un objeto que no sean externamente visibles no modificarán su valor.

Las operaciones de acceso son operaciones que leen o escriben atributos o enlaces de un objeto. No es necesario enumerarlos o especificarlos durante el análisis. Durante el diseño, es necesario observar cuáles de las operaciones de acceso van a ser públicas, y cuáles privadas para esa clase de objetos. La razón para restringir el acceso no es una razón de corrección lógica, sino más bien para encapsular las clases con el objetivo de protegerlas contra errores y para permitir modificaciones en la implementación en un futuro. Las operaciones de acceso se derivan directamente de los atributos y asociaciones de la clase dentro del modelo de objetos.

Las operaciones no triviales se pueden desglosar en tres categorías: consultas, acciones y actividades. Una consulta es una operación que carece de efectos

laterales en el estado visible externamente de cualquier objeto; es una función pura. Una consulta sin parámetros recibe el nombre de atributo derivado porque tiene la forma de un atributo.

Una acción es una transformación que posee efectos laterales sobre el objeto destino, o sobre otros objetos del sistema que resulten alcanzables desde él. Las acciones no tienen una duración temporal: son instantáneas. Dado que el estado de un objeto queda definido por sus atributos y enlaces, todas las acciones deben de ser definibles en términos de actualizaciones de los atributos y enlaces básicos. Se puede definir una acción en términos del estado del sistema antes y después de la acción, no es necesario un componente de control.

Las acciones se pueden describir de diferentes maneras, incluyendo las ecuaciones matemáticas, árboles, tablas de decisión, enumeración de todas las posibles entradas, cálculo de predicados y lenguaje natural. Es importante que la especificación sea clara y no ambigua. Es necesaria una especificación formal. Hay varios elementos en la especificación: el nombre de la función, las entradas y salidas, las transformaciones de valores y las restricciones.

Una actividad es una operación hecha por o sobre un objeto que tiene una cierta duración temporal, por oposición a las consultas y acciones, que se consideran instantáneas. Una actividad tiene efectos colaterales como consecuencia de su duración temporal. Las actividades sólo tienen sentido para actores y objetos que generen operaciones propias, porque los pasivos son solamente depósitos de datos. Los detalles de una actividad son especificados por el modelo dinámico, así como por el modelo funciones, y no se pueden considerar tan sólo una transformación. En la mayoría de los casos, una actividad corresponde a un diagrama de estados del modelo dinámico.

2.6.11.38. Restricciones

Rumbaugh (2010). Una restricción muestra la relación entre dos objetos al mismo tiempo o bien entre distintos valores del mismo objeto en instantes

diferentes. Las restricciones se pueden expresar como una función total (un valor que es especificado completamente por otro) o como una función parcial (un valor que está restringido, pero no completamente especificado por otro valor). Las restricciones pueden aparecer en todas las clases del modelo.

Las restricciones de objetos especifican que algunos objetos dependen entera o parcialmente de otros objetos. Las restricciones dinámicas especifican relaciones entre los estados o sucesos de distintos objetos. Las restricciones funcionales especifican limitaciones aplicables a operaciones.

Una restricción entre valores de un objeto a lo largo del tiempo es lo que suele denominarse un invariante.

2.6.11.39. Construcción de un modelo funcional

Rumbaugh (2010).

- Identificar valores de entrada y salida.
- Usar diagramas de flujo de datos para mostrar dependencias funcionales.
- Describir las funciones.
- Identificar restricciones.
- Especificar criterios de optimización.

2.6.11.40. Fase de Análisis.

Rumbaugh (2010). El objetivo del análisis es desarrollar un modelo del funcionamiento del sistema. El modelo se expresa en términos de objetos y relaciones, el control dinámico de flujo y las transformaciones funcionales. El proceso de capturar los requerimientos y consultar con el solicitante debe ser continuo a través del análisis. A saber:

1. Contar con una descripción inicial del problema (enunciado del problema).
2. Construir un modelo de objetos. Modelo de objetos = diagramas del

modelo de objetos + diccionario de datos.

3. Desarrollar un modelo dinámico. Modelo dinámico = diagramas de estado + diagrama global de flujo de eventos.
4. Construir un modelo funcional. Modelo funcional = diagramas de flujo de datos + restricciones.
5. Verificar, iterar y refinar los tres modelos:
 - Agregar al modelo de objetos operaciones clave que sean descubiertas durante la preparación del modelo funcional. No deben mostrarse todas las operaciones durante el análisis, sólo las más importantes.
 - Verificar que las clases, asociaciones, atributos y operaciones sean consistentes y completos al nivel seleccionado de abstracción. Comparar los tres modelos con el enunciado del problema y el conocimiento relevante al dominio y probar los modelos usando varios escenarios.
 - Desarrollar escenarios más detallados (incluyendo condiciones de error) como variaciones de los escenarios básicos, para verificar aún más los tres modelos.
 - Iterar los pasos anteriores según sea necesario para completar el análisis.

Documento de análisis = enunciado del problema + modelo de objetos + modelo dinámico + modelo funcional.

2.6.11.41. Fase de Diseño de sistemas.

Rumbaugh (2010). Durante el diseño de sistemas, se selecciona la estructura de alto nivel del sistema. Existen varias arquitecturas canónicas que pueden servir como un punto de inicio adecuado. El paradigma orientado a objetos no introduce vistas especiales en el diseño del sistema, pero se incluye para tener una cobertura completa del proceso de desarrollo de software. Los pasos son:

1. Organizar el sistema en subsistemas.
2. Identificar la concurrencia inherente al problema.
3. Asignar subsistemas a procesadores y tareas.
4. Escoger la estrategia básica para implantar los almacenamientos de

datos en términos de estructuras de datos, archivos y bases de datos.

5. Identificar recursos globales y determinar los mecanismos para controlar su acceso.
6. Seleccionar un esquema para implantar el control del software:
 - Usar la ubicación dentro del programa para mantener el estado,
 - o implantar directamente una máquina de estado,
 - o usar tareas concurrentes.
1. Considerar las condiciones de frontera.
2. Establecer prioridades de decisión sobre características deseables del producto de software.

Documento de diseño de sistemas = estructura de la arquitectura básica del sistema + las decisiones de estrategias de alto nivel.

2.6.11.42. Fase de Diseño de objetos.

Rumbaugh (2010). Durante el diseño de objetos se elabora el modelo de análisis y se proporciona una base detallada para la implantación. Se toman las decisiones necesarias para realizar un sistema sin entrar en los detalles particulares de un lenguaje o base de datos particular. El diseño de objetos inicia un corrimiento en el enfoque de la orientación del mundo real del modelo de análisis hacia la orientación en la computadora requerida para una implantación práctica. Los pasos son:

1. Obtener las operaciones para el modelo de objetos a partir de los otros modelos:
 - Encontrar una operación para cada proceso en el modelo funcional.
 - Definir una operación para cada evento en el modelo dinámico, dependiendo de la implantación del control.
2. Diseñar los algoritmos para implantar las operaciones:
 - Escoger los algoritmos que minimicen el costo de implementación de las operaciones.

- Seleccionar las estructuras de datos apropiadas para los algoritmos.
 - Definir clases internas y operaciones nuevas según sea necesario.
 - Asignar las responsabilidades para las operaciones que no están asociadas claramente con una sola clase.
3. Optimizar las rutas de acceso a los datos:
 - Agregar asociaciones redundantes para minimizar los costos de acceso y maximizar la conveniencia.
 - Reacomodar los cálculos para una mayor eficiencia.
 - Guardar los valores derivados para evitar recalcular expresiones complicadas.
 4. Implantar el control del software introduciendo el esquema seleccionado durante el diseño de sistemas.
 5. Ajustar la estructura de clases para incrementar la herencia:
 - Reacomodar y ajustar las clases y las operaciones para incrementar la herencia.
 - Abstraer el comportamiento común de los grupos de clases.
 - Usar delegación para compartir comportamiento donde la herencia sea semánticamente inválida.
 6. Diseñar la implantación de las asociaciones:
 - Analizar las travesías de las asociaciones.
 - Implantar cada asociación como un objeto distinto o agregando atributos objeto-valor a una o ambas clases en la asociación.
 7. Determinar la representación de los atributos de los objetos.
 8. Empaquetar las clases y las asociaciones en módulos.

Documento de diseño = modelo de objetos detallado + modelo dinámico detallado + modelo funcional detallado.

2.7. Servicios Generales en la Empresa Moderna

2.7.1. Introducción

Pulgar (2008). Los SG en la empresa moderna, cobra cada día mayor importancia, hasta hace unos años era considerado la cenicienta de la

organización, pero cada día se hace más indispensable para el funcionamiento de las empresas, manufactureras, prestadoras de servicio y gobierno, grandes o medianas, hasta el punto que se hace necesario elevarla de nivel, ya que es el apoyo logístico para las actividades de los distintos departamentos de la organización.

Como también es cierto este departamento crece según va creciendo la organización, ¿pero de qué forma ha crecido?, ¿Cómo se han formado los departamentos de SG?, ¿Quién ha dirigido estos departamentos?, ¿Por qué surge la necesidad de hacerlos más eficientes?, ¿Por qué se hace necesario capacitar al personal de SG?, ¿en que contribuye los SG al funcionamiento óptimo de la empresa y a reducir gastos?,

Estas y otras preguntas nos hacemos quienes hemos tenido esta responsabilidad de dirigir este departamento en la organización, y nos hemos encontrado con el desastre administrativo más grande, que nadie sabe nada, ni donde está, que fue lo que se hizo o se dejó de hacer, donde está la información de los proveedores de servicios, facturas y pare de contar, además por ser la cenicienta también los recursos para nuestra parte administrativa también son escasos ya que para los demás entes estos no son necesarios , pero a la hora de pedir cuentas salimos muy mal parados, y trabajamos con las uñas, hay muchos casos que ni una computadora aunque sea viejita contamos, tampoco en los desarrolladores de software administrativos aparecemos, y entre otras cosas amigo lector las cuales ha vivido en este departamento, además de llevar sobre los hombros un peso grande de que todo lo malo el culpable es SG.

Bueno podemos definir a los SG como el apoyo de logístico de la empresa, organización, fábrica, etc. El cual interactúa con el cliente final (llámese cliente final a todos los miembros que conforman la organización, y visitantes) de manera de garantizar que el ambiente de trabajo se desarrolle en forma organizada, eficiente, en armonía y estimule al trabajador a la excelencia, a la

empresa mostrar su calidad y prestigio.

Como pueden analizar en la definición anterior, los SG son un elemento importantísimo en la organización, ya que de nuestro buen trabajo dependen que los demás hagan el suyo más eficiente, sobretodo en cuidar todos los detalles, de manera que el empleado se sienta cómodo y bien atendido en sus necesidades, (ojo no confundir necesidades con caprichos), para ello se da a conocer en toda la organización cuales son las responsabilidades de los SG, de manera que hacer entender que el personal de SG, no es el que tiene que buscar el desayuno, hacer los pagos de servicios a los empleados, sacar las fotocopias etc. Es decir el sirviente de un particular, para esas actividades individuales la empresa no está pagando a ese empleado lo que representa una pérdida del recurso humano.

2.7.2. La organización de Servicios Generales

Pulgar (2008). La organización del departamento, gerencia o la definición que tiene esta actividad dentro de la organización, varía dependiendo del tamaño y del volumen de actividades propias de la empresa organismo, o sociedad, en la mayoría de las empresas este depende de la gerencia de compras, administración , recursos humanos o de mantenimiento, en escasas empresas estas tienen autonomía, pero de igual forma si somos los responsables de esta actividad debemos tener nuestro propio organigrama y definir cada una de las funciones que lo miembros que la integran.

Para desarrollar este tema podemos citar una organización típica de manera que sirva de orientación para el desarrollo de nuestro propio organigrama.

Para organizar nuestro departamento es necesario tener presente los siguientes aspectos:

Misión y visión de la compañía.

Valor y visión del departamento

Funciones y atribuciones del departamento

Áreas de acción y responsabilidades

2.7.3. Definición de las actividades de SG

Pulgar (2008). Con esta recopilación de información procederemos definir y determinar las funciones y responsabilidades del personal.

- a) El encargado del departamento: este tiene como responsabilidad del funcionamiento óptimo de todas las áreas de la compañía y mantener un ambiente adecuado para que el personal de la empresa preste su mejor servicio. Además de planificar las actividades, de mantenimiento preventivo y correctivo, contratación de servicios, control de gestión, evaluación de proveedores de servicios, supervisión de personal interno y contratado a terceros, etc. Además de implementación de políticas de seguridad física industrial, y de ahorro de recursos en la organización
- b) Jefes o supervisores : estos son los encargados de realizar, ejecutar todas las actividades programadas a su área de acción, así como llevar el control en el cumplimiento de las metas por el personal bajo su cargo (reporta al gerente o encargado del departamento)
- c) Personal ejecutor es el que esta de forma directa con los clientes internos y ejecutan la acción la cual son supervisado por los jefes de su área y su responsabilidad se limita a la acción para la cual fue asignado no pudiendo realizar otra actividad sin que sea autorizado por su supervisor.

Esto no es una receta de cocina y que la misma depende del tamaño de la organización y del movimiento de la misma, simplemente es una guía, esto se puede englobar en máximo 3 jefaturas supervisoras para le ejemplo anterior, en otras organizaciones puede llegar a más 10 y se puede manejar hasta más de 500 personas.

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

3.1. Materiales y Equipos

3.1.1. Método Analítico.

En este trabajo de investigación se aplicó el método analítico que consiste en realizar los procesos de conocimiento que se inicia por la identificación de cada una de las partes que caracterizan una realidad. De esa manera se estableció la relación causa efecto entre los elementos que componen el objetivo de la investigación por lo que se tomó en cuenta toda la información del área de Servicios Generales.

3.1.2. Método Sintético.

Este método permitió identificar todas las causas del problema en este caso del área de Servicios Generales y con ellos determina que soluciones se ofrecerían a través del sistema.

3.1.3. Materiales

Los Materiales, Equipos, Software y Hardware que se utilizarán en la investigación:

Materiales	Cantidad
CD	20
Papel	1000
Lápiz	1
Cuadernos	1
Anillados	10
Software	Cantidad
Sistema Operativo Windows XP	1
Microsoft Office 2007	1

Microsoft Visual Estudio. Net 1

Hardware	Cantidad
Computadora Portátil HP Webcam.	1
Disco Duro SAMSUNG HM321HI ATA Device.320GB SATA de 5400 RPM Memoria 1024 DDR2 de 667 MHz Grabadora y Reproductor de CD y DVDs	
Impresora HP Multifunción	1
Memory Flash	1

3.2. Tipo de Investigación

3.2.1. Proyectiva

Radicó en la preparación de una propuesta para solucionar los problemas de USG¹ a partir de los datos actuales, es decir, se obtuvo una propuesta como solución a la situación actual de la U.S.G. y se identificó los problemas existentes, dando alternativas específicas considerando la factibilidad para las mismas.

3.2.2. Investigación exploratoria

La investigación exploratoria es con frecuencia el paso inicial en una serie de estudios diseñados a fin de proveer información para la toma de decisiones. Por lo que el instrumento idóneo que se utilizó fue un cuestionario, con preguntas abiertas y cerradas, dirigidos a los jefes de la unidad o departamentos que conforman la organización, para determinar la problemática existente y buscar alternativas de mejora para la misma a través de la herramienta informática.

1 Unidad de Servicios Generales

3.2.3. Investigación Descriptiva

La investigación descriptiva se caracteriza por la enunciación clara del problema de decisión, objetivos específicos de investigación y necesidades de información detalladas. Se caracteriza, además, por un diseño de investigación cuidadosamente planeado y estructurado. Puesto que el propósito es proveer información referente a preguntas o hipótesis específicas, la investigación se debe diseñar para asegurar la exactitud del problema en la información de servicios generales para luego diseñar la propuesta.

3.2.4. Bibliográfica

Por cuanto se utilizó un extenso marco bibliográfico referente a la ingeniería de software y todo el material bibliográfico necesario para obtener una perspectiva objetiva práctica y actualizada, a efecto de poder abordar el problema en la propuesta investigativa cuyos datos se obtuvieron mediante información de, libros y documentos especializados en el tema.

3.3. Fuentes

Para cumplir con los objetivos de la investigación se utilizaron las siguientes fuentes:

3.3.1. Primarias

La Información fue proporcionada por el Administrador, empleados y usuarios de la Unidad de Servicio Generales”.

3.3.2. Secundarias

Para obtener más información se acudirá a Internet, libros, módulos que servirán para la elaboración del sistema.

3.4. Técnicas e Instrumentos de Evaluación

La técnica que se utilizaron para la elaboración del sistema de control para la Unidad de servicios Generales son las siguientes:

3.4.1. Entrevista

Se realizó un listado de preguntas con la finalidad de recopilar información necesaria para realizar los procesos del trabajo, las cuales fueron tomadas en cuenta a las siguientes personas.

Ing. Víctor Piñeiro	Coordinador.
Sr. Tóala Vaque	Jefe de Bodega.
Técnicos y operarios	Plomero, Gasfitero, Climatización, etc.

3.4.2. Encuesta

Los datos que se obtuvieron fueron de un conjunto de preguntas necesarias para llegar a la obtención de los datos requeridos en la investigación.

3.4.3. Técnicas e Instrucciones de Evaluaciones.

Las técnicas e instrumentos que se utilizaron para la elaboración de este sistema, se basaron en la entrevista que se realizó al encargado del área de la Unidad de Servicios Generales.

3.5. Procedimiento Metodológico

Para el análisis y desarrollo del sistema propuesto en esta tesis se utilizó la metodología de desarrollo de software orientado a objetos, para especificar el análisis y diseño de una aplicación esta metodología incluye varios diagramas:

- Diagrama de casos de Uso
- Diagrama de secuencia
- Diagrama actividad
- Diagrama de clase

3.5.1. Concepción

En esta primera fase del proyecto el ciclo de vida, acerca del entendimiento del propósito y objetivos y obteniendo suficiente información para confirmar que el proyecto debe hacer. El objetivo de ésta fase es capturar las necesidades de los clientes en los objetivos del ciclo de vida para el proyecto.

3.5.2. Elaboración

Es la segunda fase del ciclo de vida del OpenUP se trata los riesgos significativos para la arquitectura. El propósito de esta fase es establecer la base la elaboración de la arquitectura del sistema.

3.5.3. Construcción

Esta fase está enfocada al diseño, implementación y prueba de las funcionalidades para desarrollar un sistema completo. El propósito de esta fase fue completar el desarrollo del sistema basado en la Arquitectura definida.

3.5.4. Transición

Es la última fase, cuyo propósito fue asegurar que el sistema fue entregado a

los usuarios, y se evalúa la funcionalidad y ejecución, es la última de la fase de construcción entregable.

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1. Situación actual de la Unidad de Servicios Generales

La Universidad Técnica Estatal de Quevedo cuenta con la Unidad de Servicios Generales, la cual fue creada para mantener el orden de limpieza y programar el servicio de mantenimiento de todos los bienes muebles e inmuebles de los diferentes establecimientos universitarios.

Cuenta con un personal técnico especializado en diferentes especialidades como son: Técnico Electricista, Gasfitero, Plomero, Albañil, Climatización, Pintura y Jardinería lo que permite realizar las actividades requeridas.

Cada uno de ellos se responsabiliza por su labor encomendada, permitiendo el bienestar de la institución universitaria.

Todos los materiales que ingresan y egresan son registrados en tarjetas kardex y en hojas con formato elaboradas por ellos, además llevan los registros en Microsoft Excel, teniendo en cuenta que no permite almacenar grandes cantidades de información, de igual manera al momento de realizar los reportes e informes solicitados por alguna autoridad, el envío del mismo es demoroso causando molestias.

De igual manera en el departamento administrativo se observa una deficiencia por cuanto la información requerida no está a disposición, es decir, debe buscar la información en archivadores y carpetas causando gran pérdida de tiempo.

Luego de analizar la situación actual de la U.S.G., se pudo percibir que el departamento administrativo y bodega están siendo afectada por la falta de automatización por tal motivo requería la necesidad de realizar un sistema de información administrativo que cumpla con las exigencias de dicho departamento y mejorar la toma de decisiones y entrega de reportes a tiempo.

4.1.1. Requisitos

En esta primera etapa de desarrollo del software se determinaron los requisitos del sistema, es decir los requerimientos que se cumplieron para la elaboración del sistema de manera que cumpla con las necesidades de los usuarios.

4.1.2. Análisis

En esta etapa se establecieron con exactitud cuáles fueron las características del sistema y también se diseñaron los diagramas requeridos que se utilizaron en esta metodología de desarrollo con la finalidad de cumplir con los requerimientos a seguir.

4.1.3. Diseño Preliminar.

En el diseño preliminar se estableció la arquitectura del sistema. Misma que consiste en un esquema de los módulos en los que se dividió el sistema, y las librerías que se utilizaron. Considerando los casos de usos, se tomó en cuenta lo que es general, lo que específico y lo realizable.

4.1.4. Diseño Detallado.

En el diseño detallado se describieron a nivel de clases y métodos. Por cada módulo que extraído en el paso anterior y siguiendo siempre los casos de uso, se fueron detallando las clases que se implementaron y los métodos incluidos en ellas. Se desarrollaron aún más los casos de uso y las interfaces de las clases.

4.1.5. Implementación y Pruebas.

Lo principal en esta etapa es el desarrollo o implementación de la aplicación, aquí se codificaron según lo diseñado para cumplir con los requisitos

determinados en las primeras fases. De las pruebas se indican que hay que escribir los "casos de prueba". Básicamente son como la descripción de los casos de uso, pero indicando datos concretos que el operador introdujo y qué resultados exactos proporcionó el sistema.

4.2. Análisis e Interpretación de Resultados

Se realizaron 11 encuestas al personal que labora en la Unidad de servicios Generales y a 50 encuestas al personal que recibe el servicio de la U.S.G.

4.2.1. Resultados de las encuesta a los usuarios que reciben los servicios de la Unidad de Servicios Generales de la UTEQ

4.2.1.1. Calificación de la atención a los usuarios

Referente al cuadro # 1: El 34% de los encuestados responden que la atención es regular, debido a la falta de gestión y control que agilite el proceso de atención en esta Unidad.

Con esta pregunta se observó que los procesos actuales afectan la eficiencia interna de esta Unidad, al no prestar atención rápida y oportuna a sus usuarios. Esta situación obliga a plantearse la idea de que un sistema informático es necesario para que esta dependencia llegue a la excelencia en el desempeño de sus labores.

Cuadro N° 1 Calificación de la atención a los usuarios

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Regular	17	34%
Bueno	15	30%
Muy Bueno	11	22%
Excelente	7	14%
TOTAL	50	100%

Fuente: Las Encuestas

Elaborado por: El autor

4.2.1.2. Trabajo de manera técnico y profesional.

Referente al cuadro # 2: El 70% de los encuestados responden que no se realiza el trabajo de la USG de manera técnica y profesional, como consecuencia un trabajo mal realizado ocasiona pérdida de tiempo y dinero a la Universidad.

Con esta respuesta se evidencia la falta de un sistema de gestión y control para optimizar el trabajo en forma técnica y profesional en esta dependencia.

Cuadro 2. Trabajo de manera técnico y profesional

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Si	15	30%
No	35	70%
TOTAL	50	100%

Fuente: Las Encuestas

Elaborado por: el autor

4.2.1.3. Trabajo de manera eficaz, eficiente y oportuna.

Referente al cuadro # 3: El 56% de los encuestados responden que no se realiza el trabajo de la USG de eficaz, eficiente y oportuna, lo que conlleva al retraso de las actividades que se realizan en los departamentos de la Universidad.

Con esta respuesta se observó que la delegación de trabajos de acuerdo a la capacidad y disponibilidad de sus empleados dan motivo a la quejas de los usuarios por no ser cumplidas a cabalidad con eficiencia y eficacia.

Cuadro 3. Trabajo de manera eficaz, eficiente y oportuna

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Si	22	44%
No	28	56%
TOTAL	50	100%

Fuente: Las Encuestas
Elaborado por: el autor

4.2.1.4. Está de acuerdo se automática la USG.

Referente al cuadro # 4: El 100% de los encuestados contestaron que sí están de acuerdo que en esta Unidad se implemente un sistema informático para una mejor atención y desempeño de su trabajo.

La respuesta a esta pregunta afirma la necesidad de implantar el sistema la gestión y control de los servicios que presta la USG, ya que disminuirá errores e incrementará el cumplimiento de las tareas encargadas a esta dependencia.

Cuadro 4. ¿Está de acuerdo se automatice la USG?

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Si	50	100%
No	0	0%
TOTAL	50	100%

Fuente: Las Encuestas
Elaborado por: el autor

4.2.2. Resultados de las encuesta a los empleados que laboran en la Unidad de Servicios Generales de la UTEQ

4.2.2.1. Calificación de la atención que brindan

Referente al cuadro # 5: El 14% de los encuestados responden que la atención es buena, puesto que el proceso de cumplimiento de labores se hace de acuerdo a la disponibilidad de tiempo y número de empleados.

Con esta pregunta se observó que los procesos actuales pueden mejorar para llegar a la excelencia, ya que no existe una planificación de las actividades a ellos encomendadas.

Cuadro Nº 5 Calificación de la atención que brindan

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Regular	1	2%
Bueno	7	14%
Muy Bueno	1	2%
Excelente	1	2%
TOTAL	10	20%

Fuente: Las Encuestas
Elaborado por: El autor

4.2.2.2. Problemas al realizar pedido de materiales.

Referente al cuadro # 6: El 70% de los encuestados responden que si hay problemas al realizar el pedido de materiales, por motivo de no contar un stock de los mismos.

Con esta respuesta se observó que debido a que no cuentan con un sistema que les permita almacenar toda la información y facilite ese trabajo, es difícil controlar los suministros requeridos para el buen desempeño de su trabajo

Cuadro 6. Problemas al realizar pedido de materiales.

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Si	7	70%
No	3	30%
TOTAL	10	100%

Fuente: Las Encuestas

Elaborado por: el autor

4.2.2.3. Informe de daños al asignar trabajo.

Referente al cuadro # 7: El 100% de los encuestados responden que si emiten informes de los daños encontrados en las diferentes áreas o departamentos que deben dar solución, aunque en ciertas ocasiones estos informes no detallan adecuadamente lo detectado.

Con esta respuesta se observó que la falta de un sistema de gestión y control de los servicios que presta la USG no permite obtener información relevante para la mejora de la calidad de servicio que presta esta Unidad.

Cuadro 7. Informe de daños al asignar trabajo

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Si	10	100%
No	0	0%
TOTAL	10	100%

Fuente: Las Encuestas

Elaborado por: el autor

4.2.2.4. ¿Está de acuerdo se automatice la USG?

Referente al cuadro # 8: El 100% de los encuestados responden que si están de acuerdo que se automatice la Unidad, por motivo de que se mejorará el servicio a los usuarios; la respuesta a esta pregunta afirma la necesidad de

implantar el sistema informático, ya que disminuirá errores en duplicación, y pérdida de documentos, por ende se dará mayor cumplimiento en las tareas asignadas. Se observó en mayor porcentaje que estas respuestas motivan a mejorar el proceso actual con un sistema informático.

Cuadro 8. ¿Está de acuerdo se automatice la USG?

ALTERNATIVAS	FRECUENCIA	PORCENTAJE
Si	10	100%
No	0	0%
TOTAL	10	100%

Fuente: Las Encuestas
Elaborado por: el autor

4.2.3. Resultados de las Entrevistas

4.2.3.1. Entrevista al Coordinador de la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo.

1. ¿A qué se dedica la Unidad de Servicios Generales?

La Unidad de Servicios Generales es una área de la Universidad Técnica Estatal de Quevedo cuyas funciones se relacionan al mantenimiento de las áreas de pintura, electricidad, albañilería, Gasfitería climatización y jardinería.

2. ¿Cómo está organizada administrativamente la Unidad de Servicios Generales?

La Unidad de Servicios Generales está organizada administrativamente por el Ing. Víctor Piñeiro coordinador, auxiliar administrativo, el Sr. Tóala Vaque, personal de servicio general como son:

Segundo Sarmiento, Franklin Erazo, Elvis Páez, Freddy Franco, Javier Macías, Galo Cano, Javier Carranza, Ector Reina, Neil Cercado, Daniel Alvares, Néstor Vargas, Julio Carabajo, José Alcívar, Euder Saldaña, Estifanio Mayorga

3. ¿Describa los procesos que se ejecutan en la Unidad de Servicios Generales?

Los procesos que se ejecutan en la Unidad de Servicios Generales son:
Solicitud de material, informe de actividades, informe de material, informe de stock de material, entre otros.

4. ¿Cuántas personas trabajan en esta Unidad?

En Unidad de Servicios Generales trabajan 17 personas los que realizan los trabajos de:
Gasfitero, plomero, pintura, albañil, climatización, jardinero

5. ¿En el caso de automatizar los procesos de la unidad de servicios generales cuales serían las funciones que deberían tomar en cuenta para este propósito?

Si No

Si la respuesta es afirmativa, escriba los nombres de los procesos.

Informes de trabajos realizados, stock de material y de contratación, base de datos – equipos climatización y mantenimiento

6. ¿Cuenta con algún tipo de control de stock de productos, materiales confiable en su unidad?

Si No

4.3. Discusión

De acuerdo al análisis de los resultados, se detectaron problemas en el almacenamiento de información, informes, reportes, etc. ocasionando pérdida de tiempo en los procesos de información, de manera que la U.S.G no cuenta con un sistema para poder llevar un mejor control como es de:

Llevar un control detallado de mantenimiento de los acondicionadores de aire de las distintas dependencias, de la misma manera de las diferentes actividades, entre ellas está la elaboración de informes, reporte, actividades, ingreso y egreso de los materiales que emite la U.S.G.

Por lo tanto al criterio de los encuestados y los entrevistados consideran necesario la implementación de un sistema de control administrativo para ayudar a cumplir con los requerimientos de la U.S.G. y mejorar todos los procesos y control.

Según las expresiones de Castillo (2007) asegura que los sistemas de control se aplican en esencia para los organismos vivos, las máquinas y las organizaciones. Estos sistemas fueron relacionados por primera vez en 1948 por Norbert Wiener en su obra cibernética y sociedad con aplicación en la teoría de los mecanismos de control. Un sistema de control está definido como un conjunto de componentes que pueden regular su propia conducta o la de otro sistema con el fin de lograr un funcionamiento predeterminado.

Esta afirmación la relaciona Fernández (2006) quien indica que un Sistema de información es el conjunto formal de procesos de análisis, que operando sobre una colección de datos estructurados de acuerdo a las necesidades de la empresa, recopila elabora y distribuye, la información necesaria para realizar las operaciones básicas y la toma de decisión en la empresa.

4.4. Propuesta

Open UP o proceso unificado abierto es una metodología ágil que aplica un enfoque iterativo e incremental dentro de su estructura del ciclo de vida, la misma que la hace adaptarse al desarrollo de diversos tipos de proyectos.

Las fases de desarrollo que la metodología Open UP utiliza son las siguientes:

4.4.1. Concepción

En esta primera fase se estableció la visión del proyecto, identificando el detalle de los componentes de las partes del documento, definiendo el entorno del desarrollo (incluyendo las herramientas y la infraestructura del sistema).

Al iniciar los niveles, se reunieron los requisitos y se analizaron los elementos necesarios para facilitar las prioridades y planificación, en esta fase se elaboró el planteamiento del problema, la justificación, definiendo el objetivo general y los específicos para capturar las necesidades de la Unidad de Servicios Generales y por ende detallar las especificaciones de casos de uso para crear un prototipo de la arquitectura.

4.4.2. Elaboración

En esta fase se proporcionó una estimación más precisa de tiempo de una manera realista.

Se realizaron las correcciones definidas en la fase anterior para lograr lo planificado así como los requisitos de gran importancia para poder utilizar la información como insumo para las actividades de la arquitectura y el desarrollo en la iteración actual, y la planificación para la siguiente iteración.

En esta fase se llevan a cabo los siguientes pasos:

- Definir las interface del usuario.
- Definir las responsabilidades que tendrá cada una de las interface de usuario.
- Definir la interacción entre las interface de usuario.
- Controlar el flujo de datos de las interfaces de usuario.

El objetivo principal de esta actividad fue de proponer una arquitectura que responda a las exigencias, proporcionadas para construir la funcionalidad del sistema.

4.4.3. Construcción

Como su palabra lo dice en esta fase se construyeron las interfaces del sistema, se desarrollaron los diagramas de Caso de Uso, actividad y los de secuencia así como también se trabajaron en la base de datos y el lenguaje de programación. El objetivo principal de esta actividad fue entregar el sistema ejecutable para ofrecer la calidad y funcionalidad de los requisitos especificados.

4.4.4. Transición

En esta fase de transición se realizaron las fases de funcionalidad, el rendimiento y la calidad del sistema para ser entregado al consumidor final o empresa.

4.5. Documentación del Sistema

El método UML es utilizado para visualizar, describir, construir y documentar el Sistema de Información SIST_USG para la Unidad de Servicios Generales UTEQ, describe el plano del sistema, incluyendo aspectos conceptuales tales como procesos de mantenimientos de equipos e inmuebles y funciones del sistema, aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

El modelo OpenUP y la variedad de alternativas propuestas por el lenguaje de Modelado Unificado (UML) se considera la elección metodológica para desarrollar el sistema de SIST_USG para la Unidad de Servicios Generales.

4.5.1. Casos de Uso

4.5.1.1. Caso de Uso General



Figura 1. Caso de Uso General.

4.5.1.2. Casos de Uso Expandido

4.5.1.2.1. Inicio de Sesión

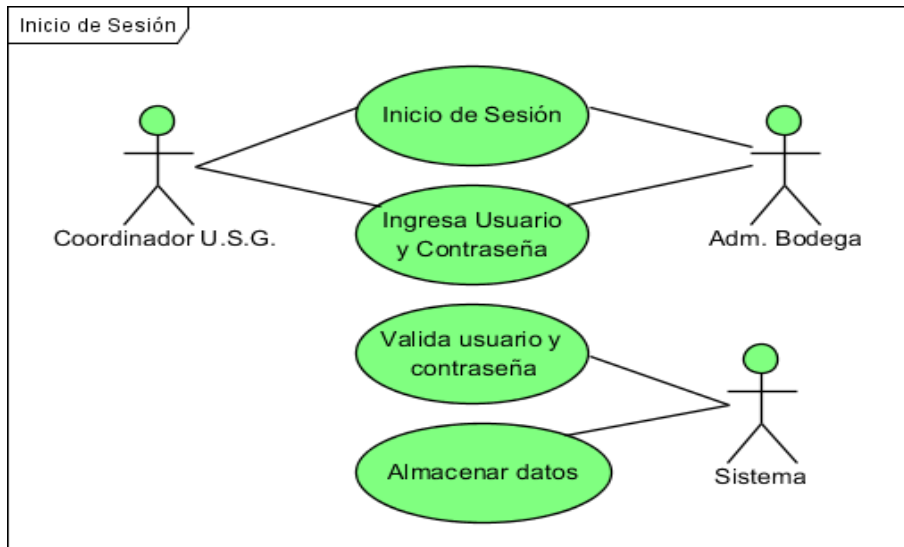


Figura 2. Caso de Uso Inicio de Sesión

4.5.1.2.2. Crear Cuenta de Usuario

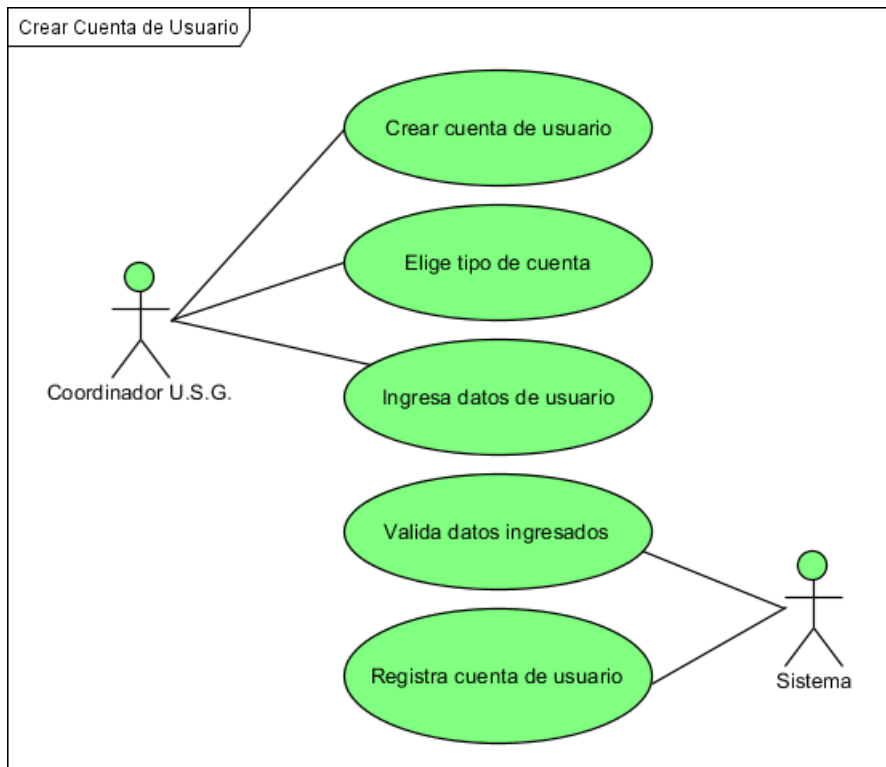


Figura 3. Crear Cuenta de Usuario

4.5.1.2.3. Registra Personal de Servicio

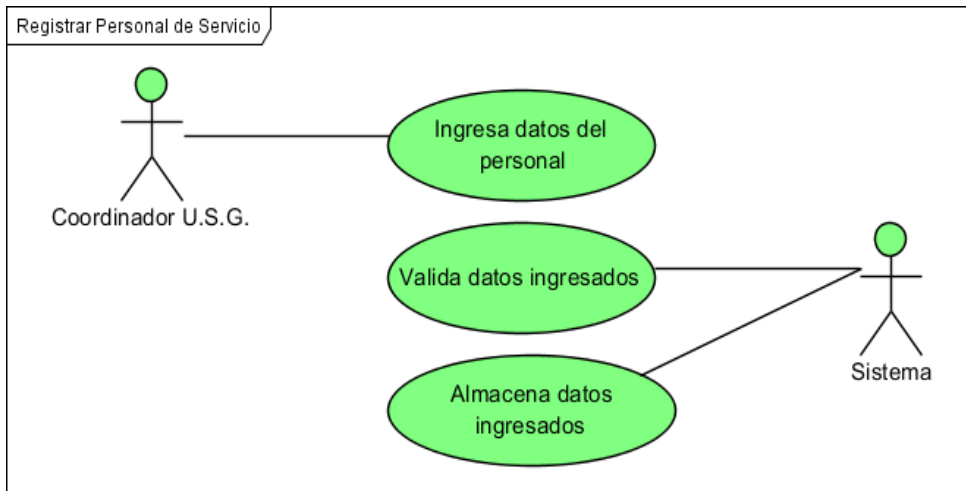


Figura 4. Caso de Uso Registral Personal de Servicio

4.5.1.2.4. Ingreso de Actividades del Personal

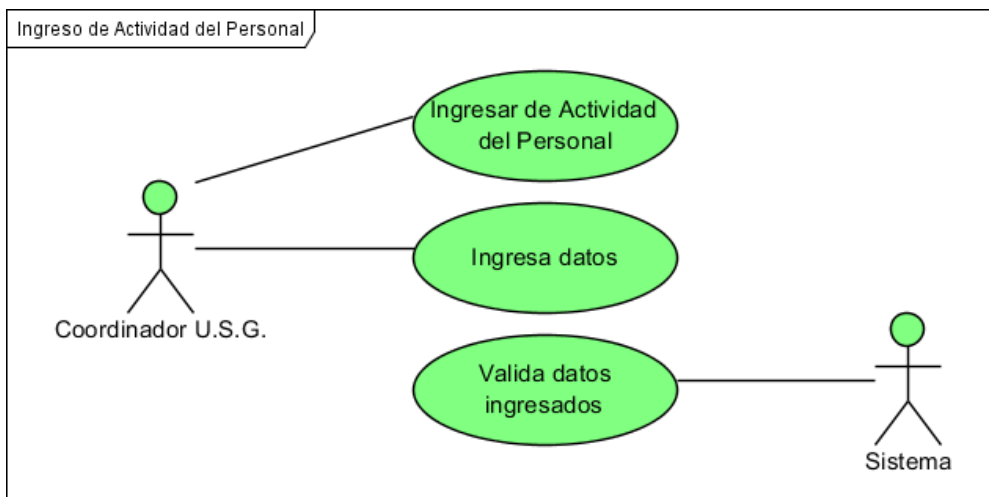


Figura 5. Caso de Uso Ingreso de Actividades del Personal

4.5.1.2.5. Ingreso de Herramienta

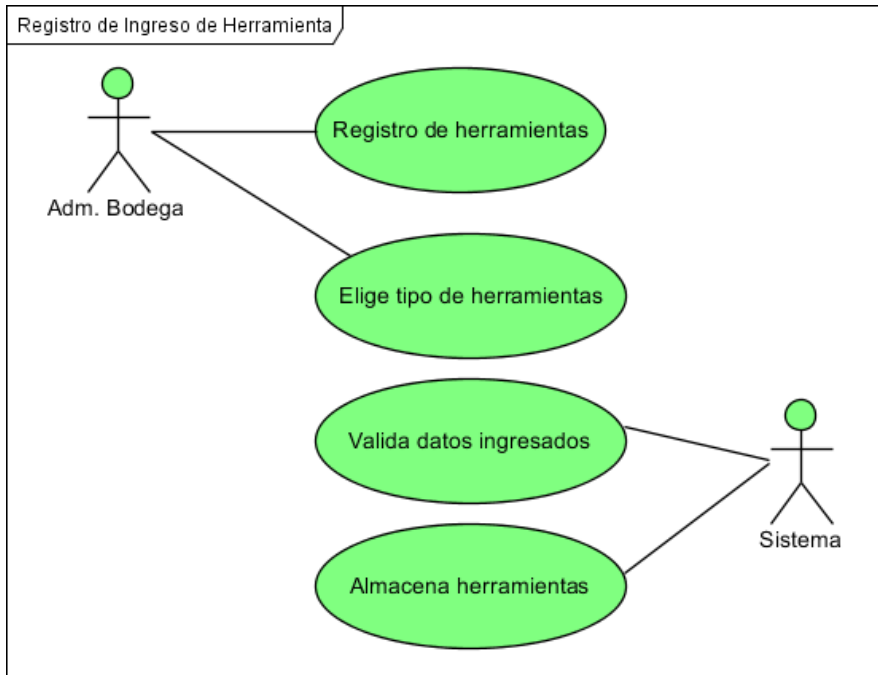


Figura 6. Caso de Uso Ingreso de Herramienta

4.5.1.2.6. Ingreso de Material

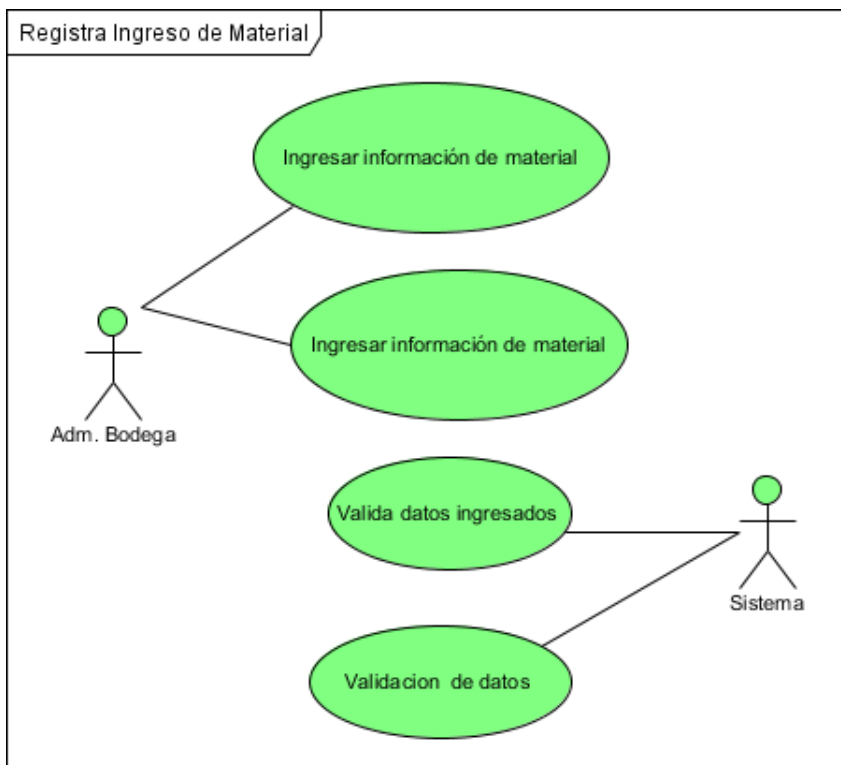


Figura 7. Caso de Uso Ingreso de Kardex

4.5.1.2.7. Registra Unidad de Medida

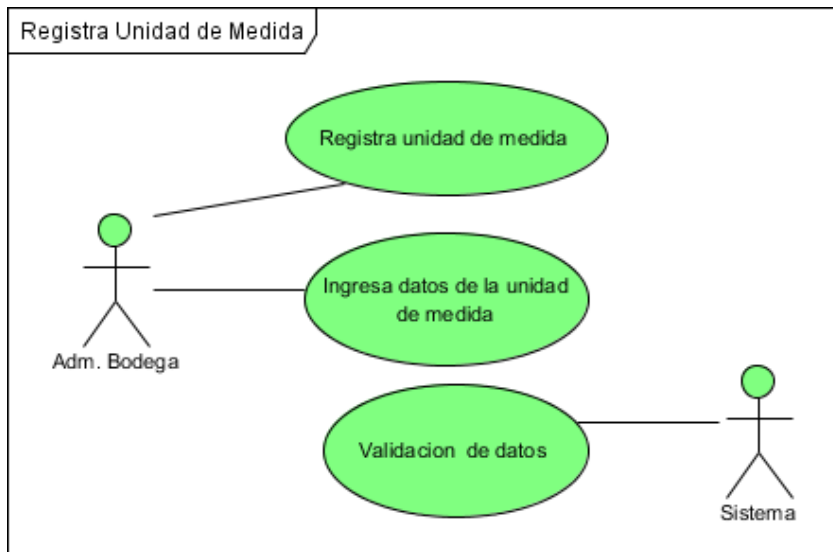


Figura 8. Caso de Uso Registra Unidad de Medida

4.5.1.2.8. Registra Ingreso de Kardex

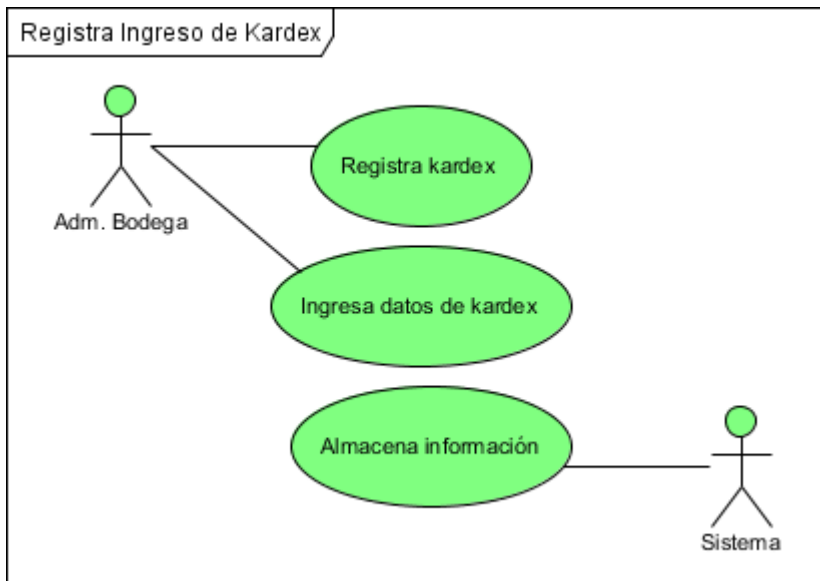


Figura 9. Caso de Uso Ingreso de Kardex

4.5.1.2.9. Realizar Consulta

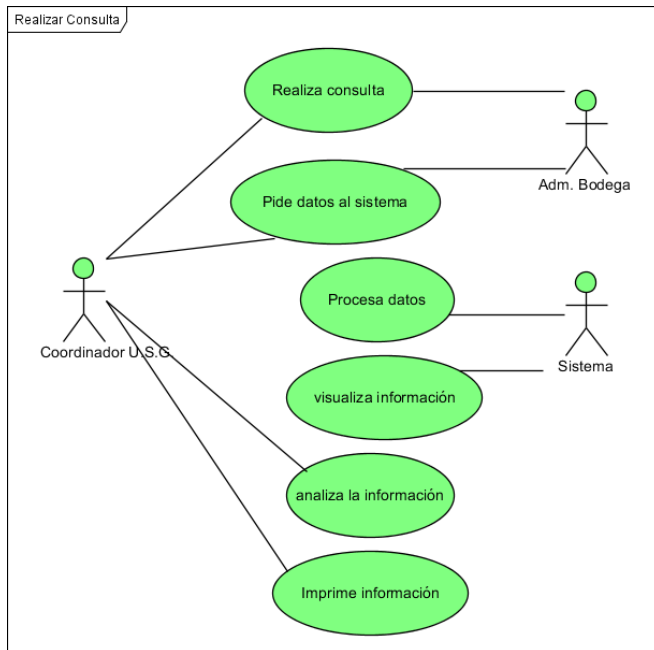


Figura 10. Caso de Uso Realizar Consulta

4.5.1.2.10. Informe de Actividades

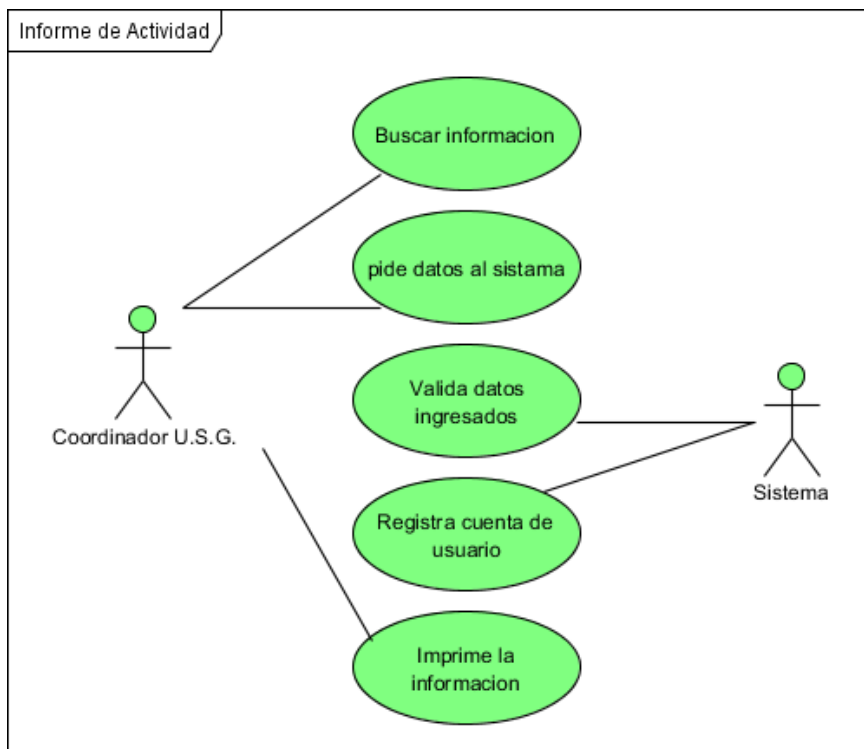


Figura 11. Caso de Uso Informe de Actividad

4.5.1.2.11. Informe de Material

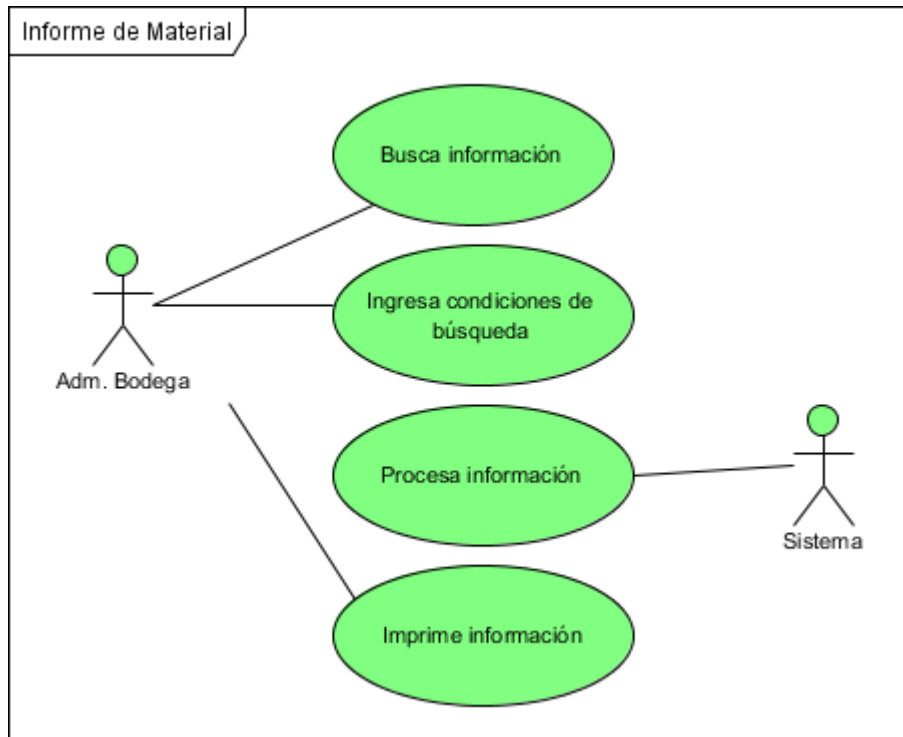


Figura 12. Caso de Uso Informe de Material

4.5.1.2.12. Salida de Material

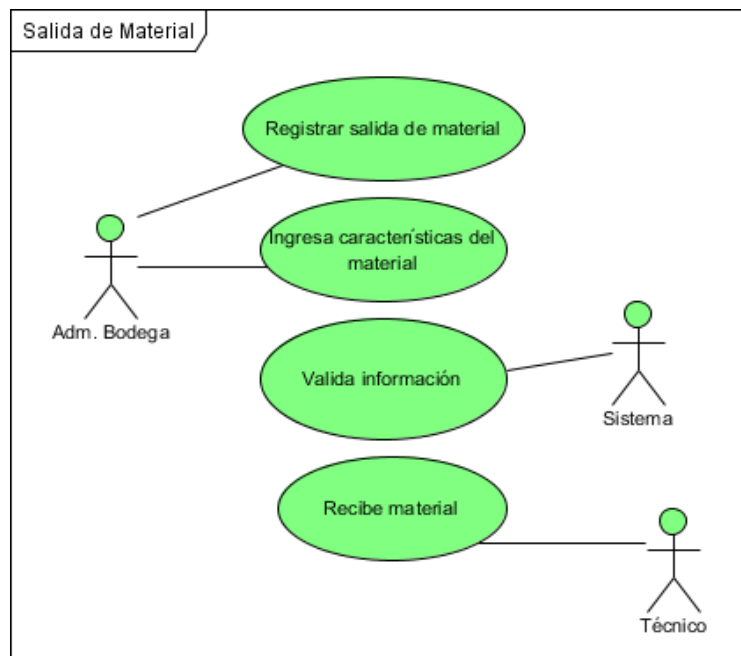


Figura 13. Caso de Uso Registra Salida de Material

4.5.1.2.13. Salida de Herramientas

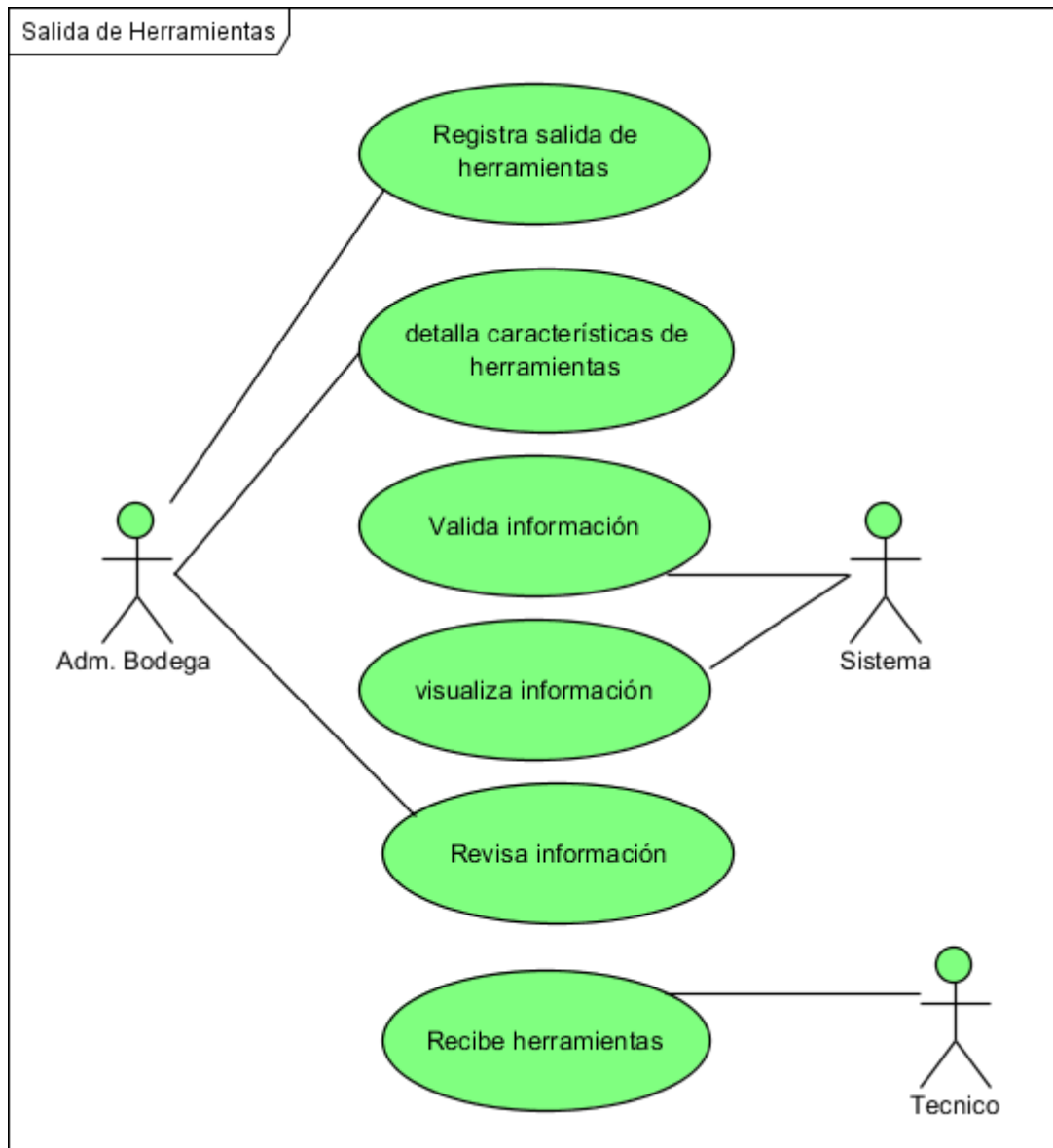


Figura 14. Caso de Uso Registra Salida de Material

4.5.2. Descripción del Caso de Uso Expandido

4.5.2.1. Inicio de Sesión

Cuadro 15. Caso de Uso Inicio de Sesión

CASO DE USO	Inicio de Sesión	
ACTOR	Coordinador de la U.S.G.	
DESCRIPCIÓN	El coordinador ingresa al sistema, escribe su login y contraseña correspondiente	
PRECONDICIONES	El Coordinador está registrado en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	1- El Coordinador ingresa su login y su contraseña en el sistema.	2- El sistema verifica los datos ingresados. 3- El sistema le permite el acceso al Coordinador.
CURSO ALTERNO	ACTOR	SISTEMA
		2.1- la contraseña del Coordinador no es correcta, y el sistema le avisa al usuario del error.
POSTCONDICIÓN	El Coordinador tiene el ingreso a todas las funciones del sistema.	

4.5.2.2. Configurar Cuenta de Usuario

Cuadro 16. Caso de Uso Configurar Cuenta de Usuario

CASO DE USO	Configurar Cuenta de Usuario	
ACTOR	Coordinador de la U.S.G.	
DESCRIPCIÓN	El coordinar registra usuario al sistema.	
PRECONDICIONES	El usuario no está registrado en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	<p>1 El Coordinador inicia sesión.</p> <p>2.- Elegí administrador.</p> <p>3.-ingresa a la opción usuario.</p> <p>5.- ingresa los datos del usuario.</p>	<p>4.- Muestra la información a ingresar.</p> <p>6.- Guarda los datos.</p>
CURSO ALTERNO	ACTOR	SISTEMA
	<p>3.1.- El administrador decide cancelar la acción.</p>	<p>5.1.- información ingresada no está completada, muestra mensaje de error.</p>
POSTCONDICIÓN	El Coordinador tiene el ingreso a todas las funciones del sistema.	

4.5.2.3. Registra Personal de Servicio

Cuadro 17. Caso de Uso Registra Personal de Servicio

CASO DE USO	Registra Personal de Servicio	
ACTOR	Coordinador de la U.S.G.	
DESCRIPCIÓN	El coordinar registra al personal de servicio en el sistema.	
PRECONDICIONES	El personal no está registrado en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El Coordinador inicia sesión. 2.- Elegí nómina de personal. 3.- Ingresa datos del personal.	4.- Muestra la información a ingresar. 5.- Guarda los datos ingresados.
CURSO ALTERNO	ACTOR	SISTEMA
	5.1.- El administrador decide salir de la aplicación.	3.1.- información ingresada no está completada, muestra mensaje de error.
POSTCONDICIÓN	El Coordinador tiene el ingreso a todas las funciones del sistema.	

4.5.2.4. Ingreso de Actividad

Cuadro 18. Caso de Uso Ingreso de Actividad

CASO DE USO	Ingreso de Actividad	
ACTOR	Coordinador de la U.S.G.	
DESCRIPCIÓN	El coordinar registra actividad en el sistema.	
PRECONDICIONES	La actividad no está registrada en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	<p>1 El Coordinador inicia sesión.</p> <p>2.- Elegí formulario de actividad.</p> <p>3.- Ingresa datos de la actividad.</p>	<p>4.- Muestra formulario a ingresar.</p> <p>5.- Guarda los datos ingresados.</p>
CURSO ALTERNO	ACTOR	SISTEMA
	<p>5.1.- El administrador decide salir de la aplicación.</p>	<p>3.1.- información ingresada no está completada, muestra mensaje de error.</p>
POSTCONDICIÓN	El Coordinador tiene el ingreso a todas las funciones del sistema.	

4.5.2.5. Registra Tipo de Herramienta

Cuadro 19. Caso de Uso Ingreso de Actividad

CASO DE USO	Ingreso de Actividad	
ACTOR	Coordinador de la U.S.G.	
DESCRIPCIÓN	El coordinar registra actividad en el sistema.	
PRECONDICIONES	La actividad no está registrada en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	<p>1 El Coordinador inicia sesión.</p> <p>2.- Elegí formulario de actividad.</p> <p>4.- Ingresa datos de la actividad.</p>	<p>3.- Muestra formulario a ingresar.</p> <p>5.- Guarda los datos ingresados.</p>
CURSO ALTERNO	ACTOR	SISTEMA
	<p>5.1.- El administrador decide salir de la aplicación.</p>	<p>3.1.- información ingresada no está completada, muestra mensaje de error.</p>
POSTCONDICIÓN	El Coordinador tiene el ingreso a todas las funciones del sistema.	

4.5.2.6. Registra Tipo de Material

Cuadro 20. Caso de Uso Registra Tipo de Material

CASO DE USO	Registra Tipo de Material	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El Administrador de bodega registra material al sistema.	
PRECONDICIONES	El material no está registrado en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	1 El Administrador de bodega inicia sesión. 2.- Elegí formulario de material. 3.- selecciona tipo de material 5.- Ingresa datos de material.	4.- Muestra formulario a ingresar. 6.- Guarda los datos ingresados.
CURSO ALTERNO	ACTOR	SISTEMA
	3.1.- El Administrador de bodega decide salir de la aplicación.	5.1.- información ingresada no está completada, muestra mensaje de error.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.7. Registra Unidad de Medida

Cuadro 21. Caso de Uso Registra Unidad de Medida

CASO DE USO	Registra Unidad de Medida	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El Administrador de bodega registra unidad de medida en el sistema.	
PRECONDICIONES	La unidad de medida no está registrada en el sistema.	
CURSO NORMAL	ACTOR	SISTEMA
	<p>1 El Administrador de bodega inicia sesión.</p> <p>2.- Elegí formulario de unidad de medida.</p> <p>3.- selecciona tipo de unidad de medida</p> <p>5.- Ingresa datos de unidad de medida.</p>	<p>4.- Muestra la información requerida para ingresar.</p> <p>6.- Guarda los datos ingresados.</p>
CURSO ALTERNO	ACTOR	SISTEMA
	<p>3.1.- El Administrador de bodega decide salir de la aplicación.</p>	<p>5.1.- información ingresada no está completada, muestra mensaje de error.</p>
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.8. Registra Ingreso de Kardex

Cuadro 22. Caso de Uso Registra Ingreso de Kardex

CASO DE USO	Registra Ingreso de Kardex	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El Administrador de bodega registra kardex al sistema.	
PRECONDICIONES	El Administrador de bodega deberá ingresar su clave y cuenta de usuario.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El Administrador de bodega inicia sesión. 2.- Elegí formulario de kardex. 4.- Ingresa datos del kardex.	3.- Muestra formulario para ingresar kardex. 5.- Guarda los datos ingresados.
CURSO ALTERNO	ACTOR	SISTEMA
	2.1.- El Administrador de bodega decide salir de la aplicación.	4.1.- información ingresada no está completada, muestra mensaje de error.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.9. Realiza Consulta

Cuadro 23. Caso de Uso Realiza Consulta

CASO DE USO	Realiza Consulta	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El Administrador de bodega realiza consulta al sistema.	
PRECONDICIONES	El Administrador de bodega deberá ingresar su clave y cuenta de usuario.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El Administrador de bodega inicia sesión. 2.- Elegí opciones de formularios para realiza consulta. 4.- Realiza consulta.	3.- Muestra opciones de formularios. 5.- Imprime.
CURSO ALTERNO	ACTOR	SISTEMA
	2.1.- El Administrador de bodega decide salir de la aplicación.	4.1.- información ingresada no está completada, muestra mensaje de error.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.10. Realiza Informe de Actividad

Cuadro 24. Caso de Uso Realiza Informe de Actividad

CASO DE USO	Realiza Informe de Actividad	
ACTOR	Coordinador de la U.S.G.	
DESCRIPCIÓN	El Coordinador de la U.S.G. realiza informe de actividad al sistema.	
PRECONDICIONES	El Coordinador de la U.S.G. deberá ingresar su clave y cuenta de usuario.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El Coordinador de la U.S.G. inicia sesión. 2.- Elegí opciones e ingresa al formulario de actividad. 4.- Realiza informe de actividad.	3.- Muestra formulario de actividad. 5.- Imprime.
CURSO ALTERNO	ACTOR	SISTEMA
	2.1.- El Coordinador de la U.S.G. decide salir de la aplicación.	4.1.- informe de actividad no está actualizada.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.11. Realiza Informe de Material

Cuadro 25. Caso de Uso Realiza Informe de Material

CASO DE USO	Realiza Informe de Material	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El administrador de bodega realiza informe de material en el sistema.	
PRECONDICIONES	El administrador de bodega deberá ingresar su clave y cuenta de usuario.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El administrador de bodega inicia sesión. 2.- Elegí formulario de material. 4.- Realiza informe de material.	3.- Muestra formulario de material. 5.- Imprime.
CURSO ALTERNO	ACTOR	SISTEMA
	2.1.- El Administrador de bodega decide salir de la aplicación.	4.1.- informe de material no está actualizado.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.12. Registra Salida de Material

Cuadro 26. Caso de Uso Registra Salida de Material

CASO DE USO	Registra salida de material	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El administrador de bodega registra salida de material en el sistema.	
PRECONDICIONES	El administrador de bodega deberá ingresar su clave y cuenta de usuario.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El administrador de bodega inicia sesión. 2.- Elegí formulario de material. 4.- Registra la salida de material.	3.- Muestra formulario de material. 5.- Guarda la información.
CURSO ALTERNO	ACTOR	SISTEMA
	2.1.- El Administrador de bodega decide salir de la aplicación.	4.1.- El material se encuentra incompleto.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

4.5.2.13. Registra Salida de Herramienta

Cuadro 27. Caso de Uso Registra Salida de Herramienta

CASO DE USO	Registra salida de herramienta	
ACTOR	Administrador de bodega de la U.S.G.	
DESCRIPCIÓN	El administrador de bodega registra salida de herramientas en el sistema.	
PRECONDICIONES	El administrador de bodega deberá ingresar su clave y cuenta de usuario.	
CURSO NORMAL	ACTOR	SISTEMA
	1.- El administrador de bodega inicia sesión. 2.- Elegí formulario de herramientas. 4.- Registra la salida de las herramientas.	3.- Muestra formulario de herramientas. 5.- Guarda la información.
CURSO ALTERNO	ACTOR	SISTEMA
	2.1.- El Administrador de bodega decide salir de la aplicación.	4.1.- El material se encuentra incompleto.
POSTCONDICIÓN	El Administrador de bodega no tiene el ingreso a todas las funciones del sistema.	

Diagrama de Actividad

4.5.2.14. Inicio de Sesión

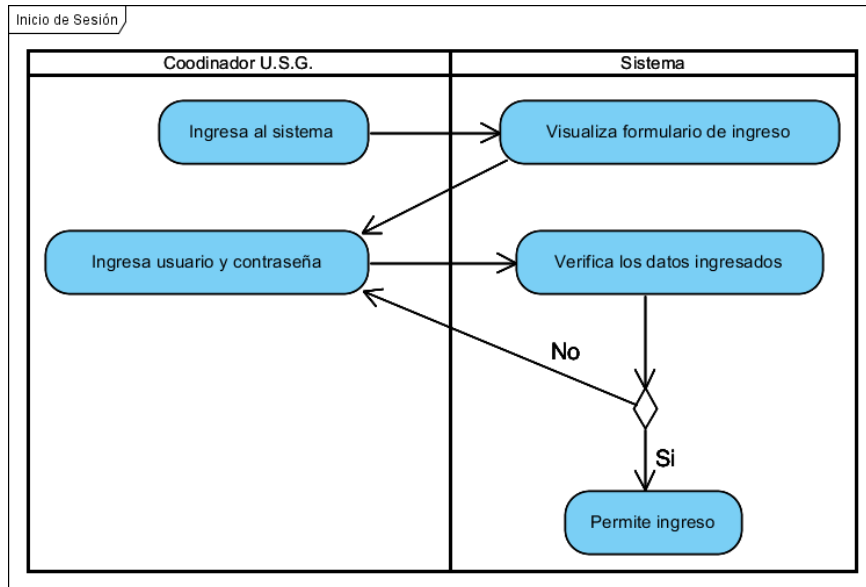


Figura 1. Inicio de Sesión

4.5.2.15. Configurar Cuenta de Usuario

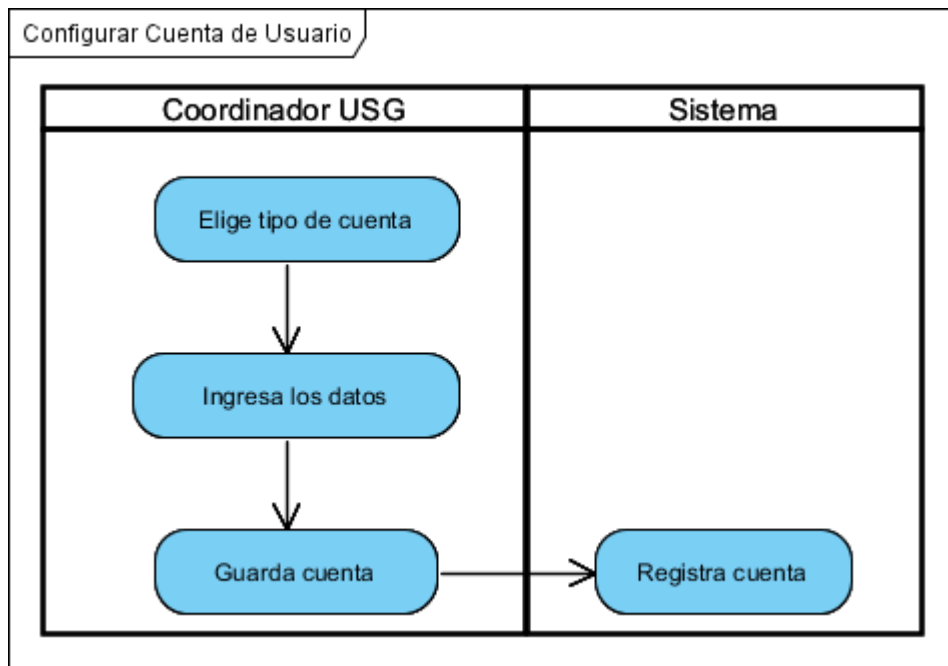


Figura 2. Configurar Cuenta de Usuario

4.5.2.16. Registra Personal de Servicio

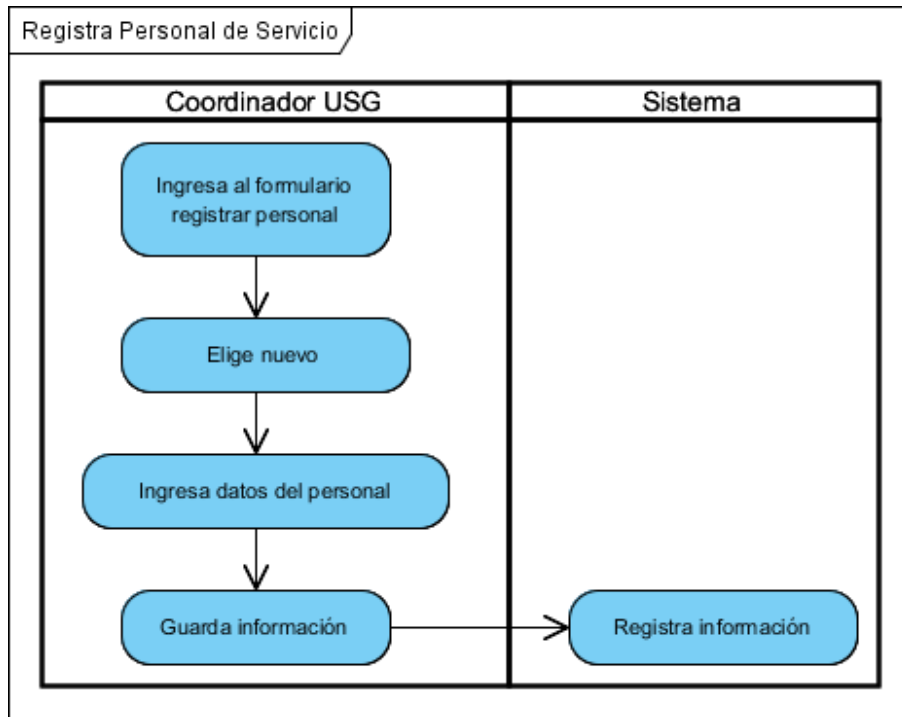


Figura 3. Registra Personal de Servicio

4.5.2.17. Ingreso de Actividades

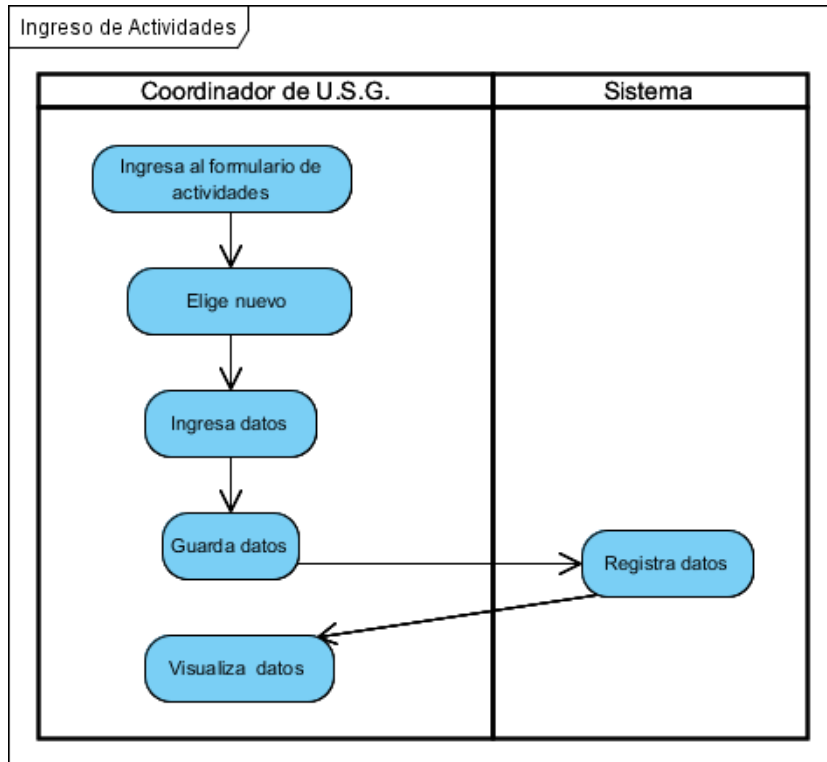


Figura 4. Ingreso de Actividades

4.5.2.18. Registra Ingreso de Herramientas

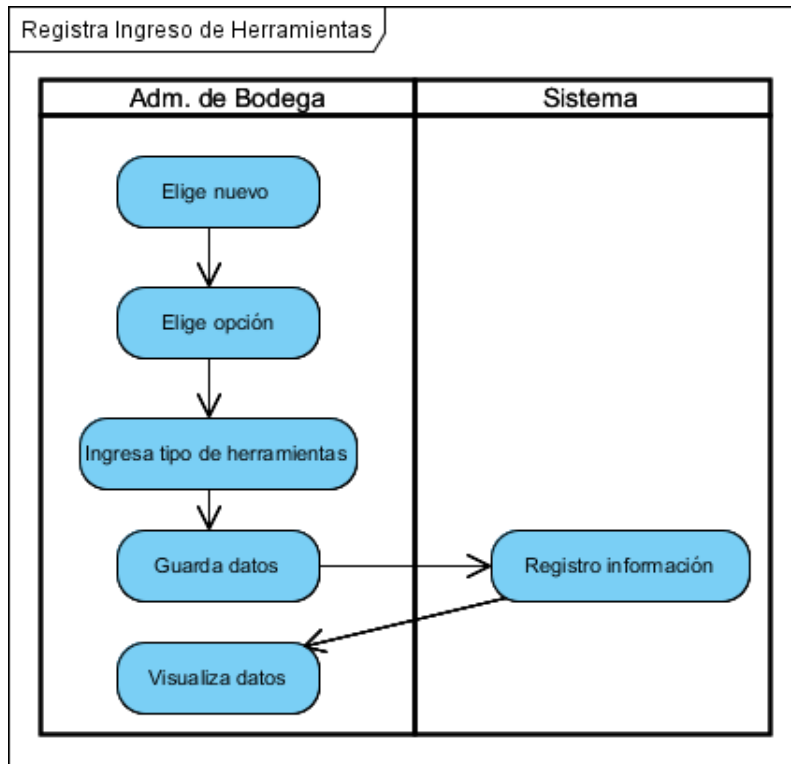


Figura 5. Registra Ingreso de Herramientas

4.5.2.19. Registra Ingreso de Material

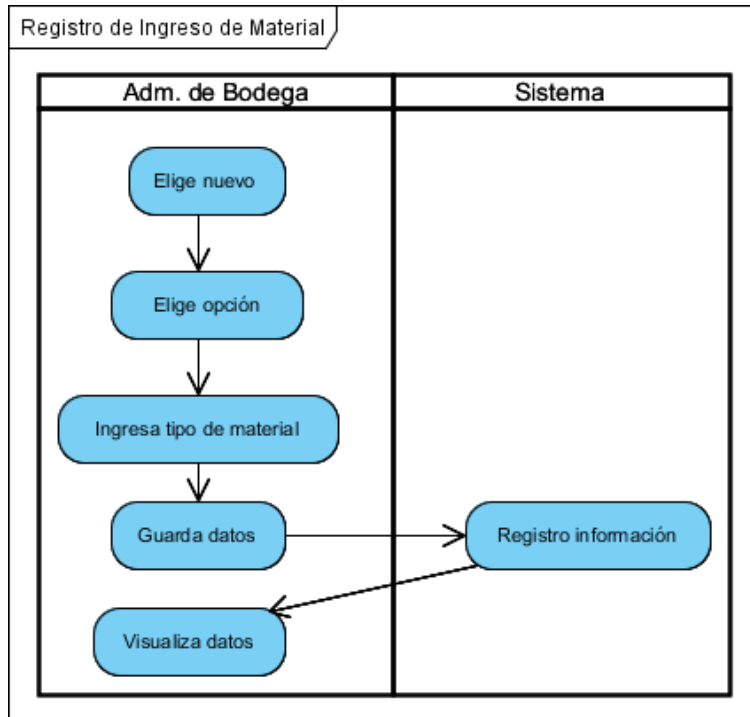


Figura 6. Registra Ingreso de Material

4.5.2.20. Registra Unidad de Medida

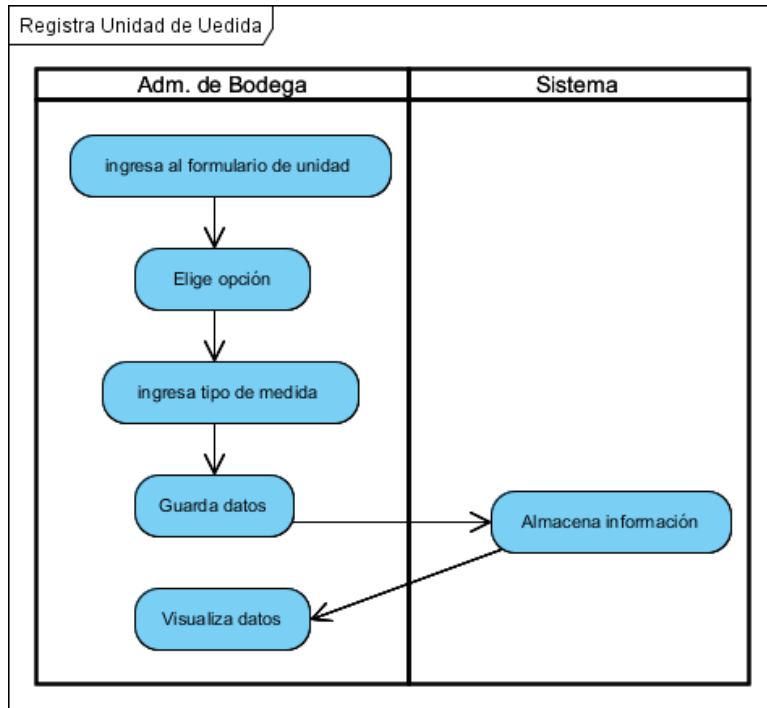


Figura 7. Registra Unidad de Medida

4.5.2.21. Ingreso de Kardex

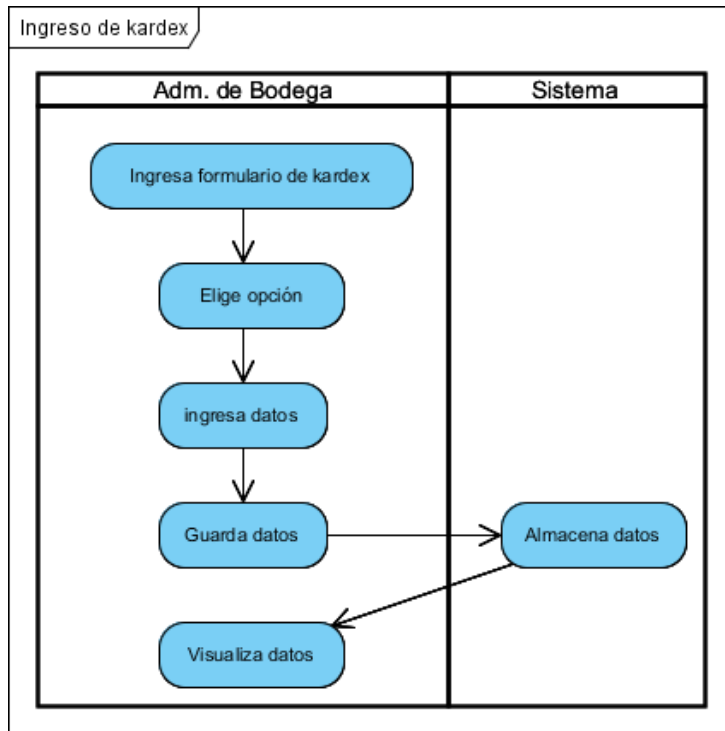


Figura 8. Registra Ingreso de Kardex

4.5.2.22. Realizar Consulta

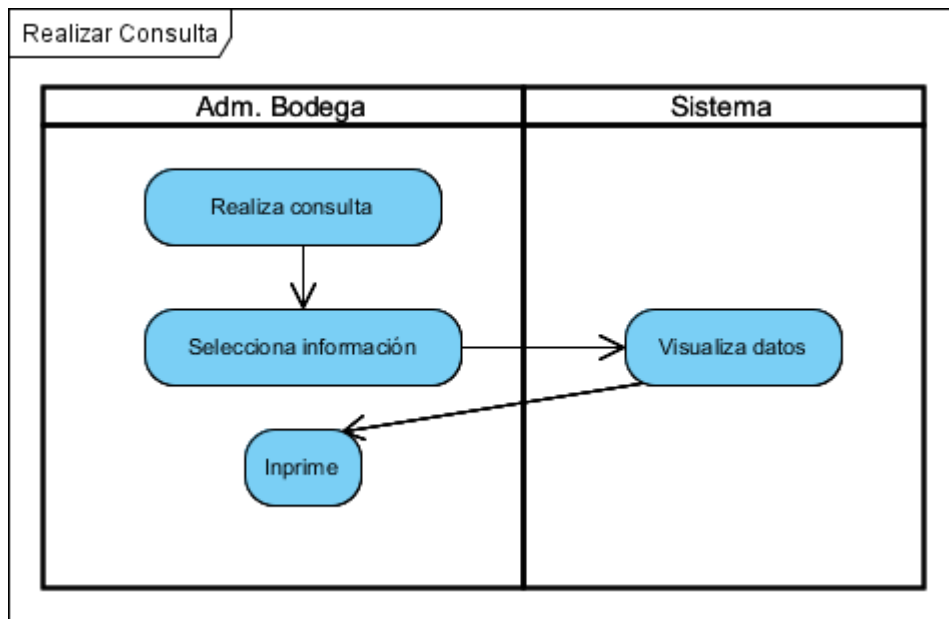


Figura 9. Realizar Consulta

4.5.2.23. Realiza Informe de Actividad

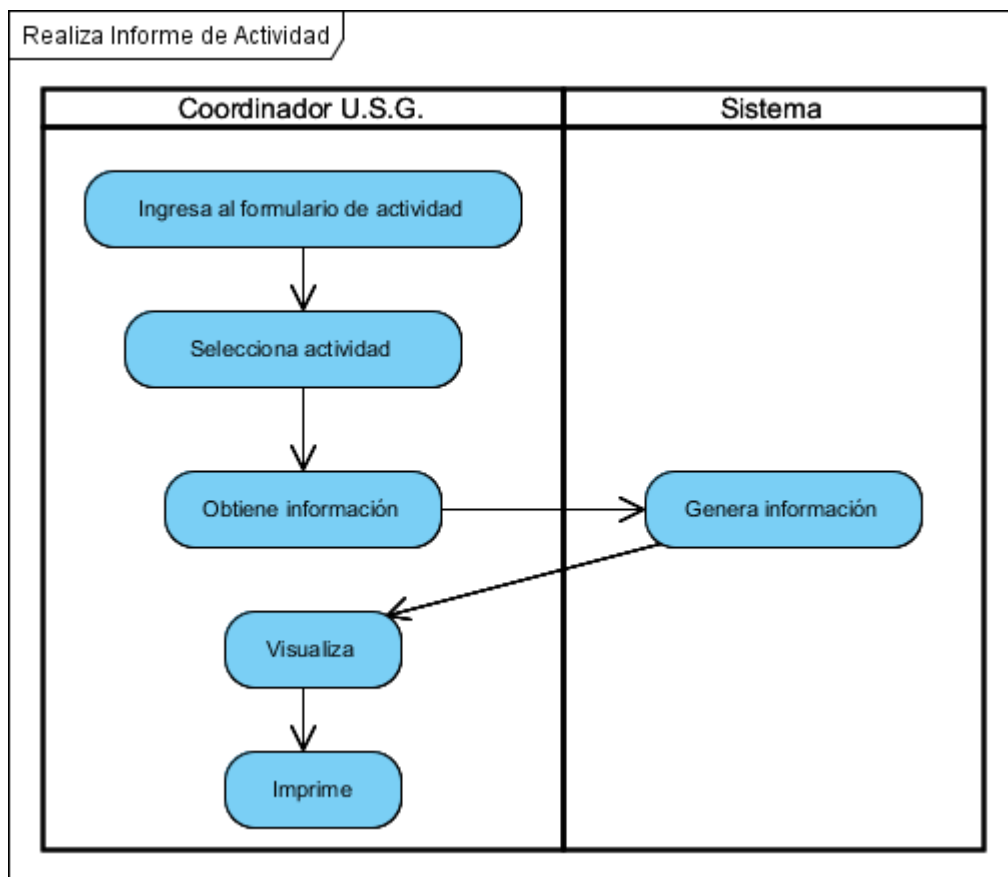


Figura 10. Realizar Informe de Actividad

4.5.2.24. Informe de Material

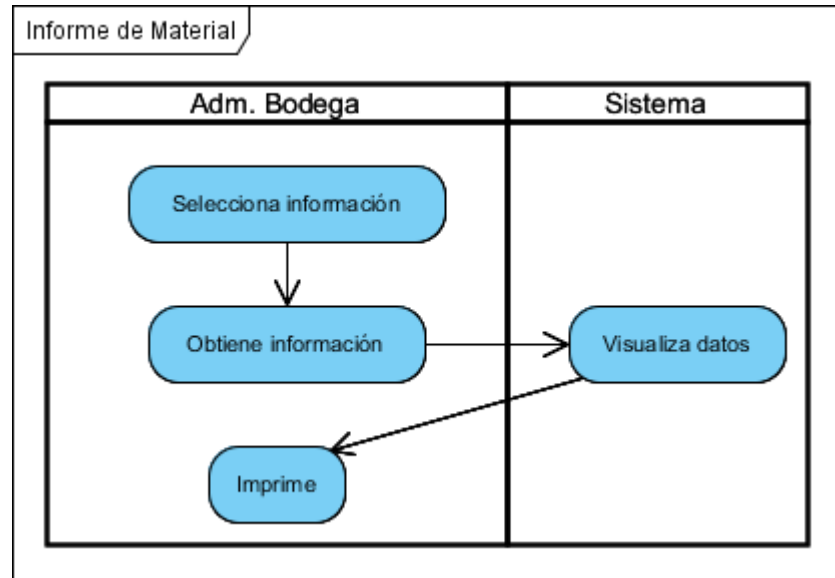


Figura 11. Informe de Material

4.5.2.25. Salida de Material

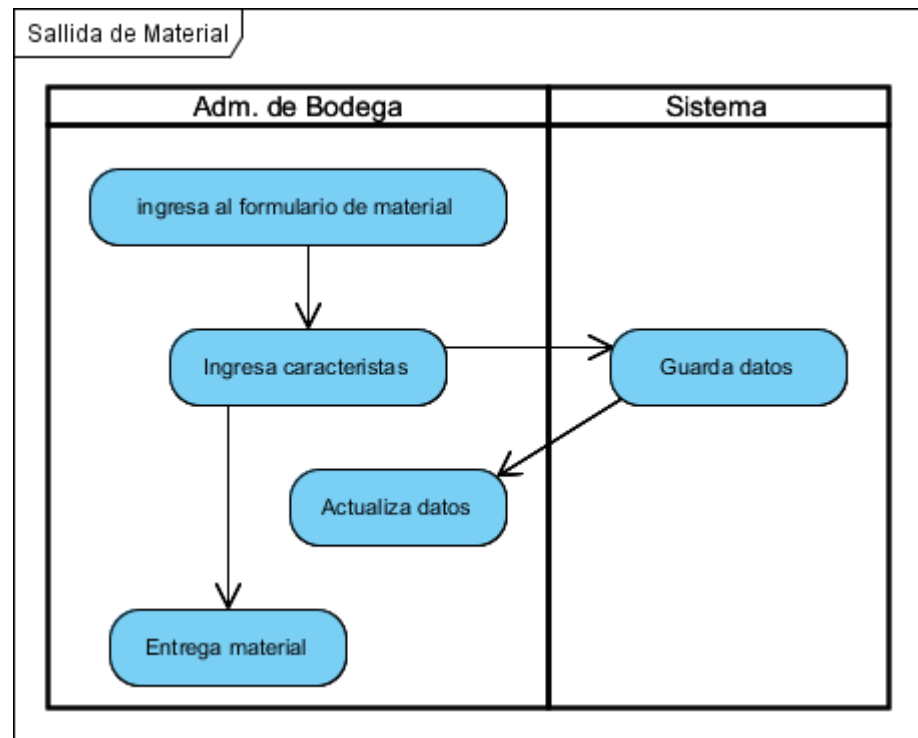


Figura 12. Salida de Material

4.5.2.26. Realiza Salida de Herramientas

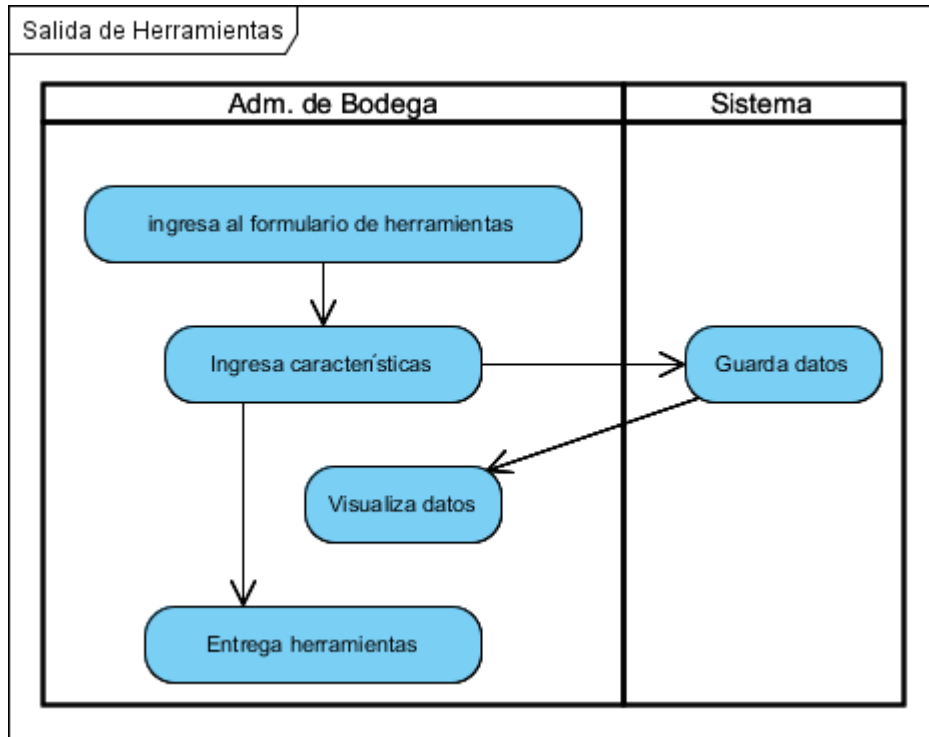


Figura 13. Salida de Herramienta

4.5.3. Diagrama de Secuencia

4.5.3.1. Inicio de Sesión

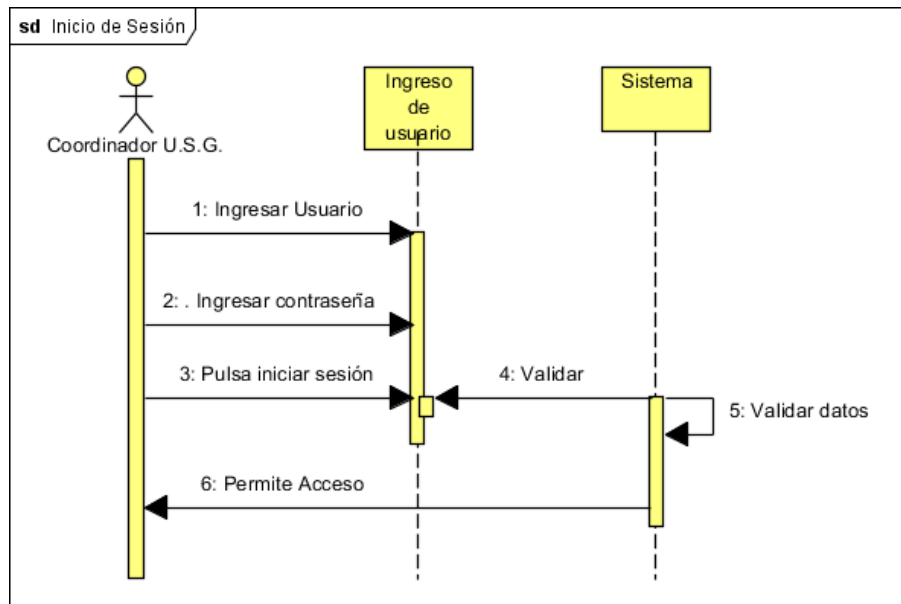


Figura 1. Inicio de Sesión

4.5.3.2. Configurar Cuenta de Usuario

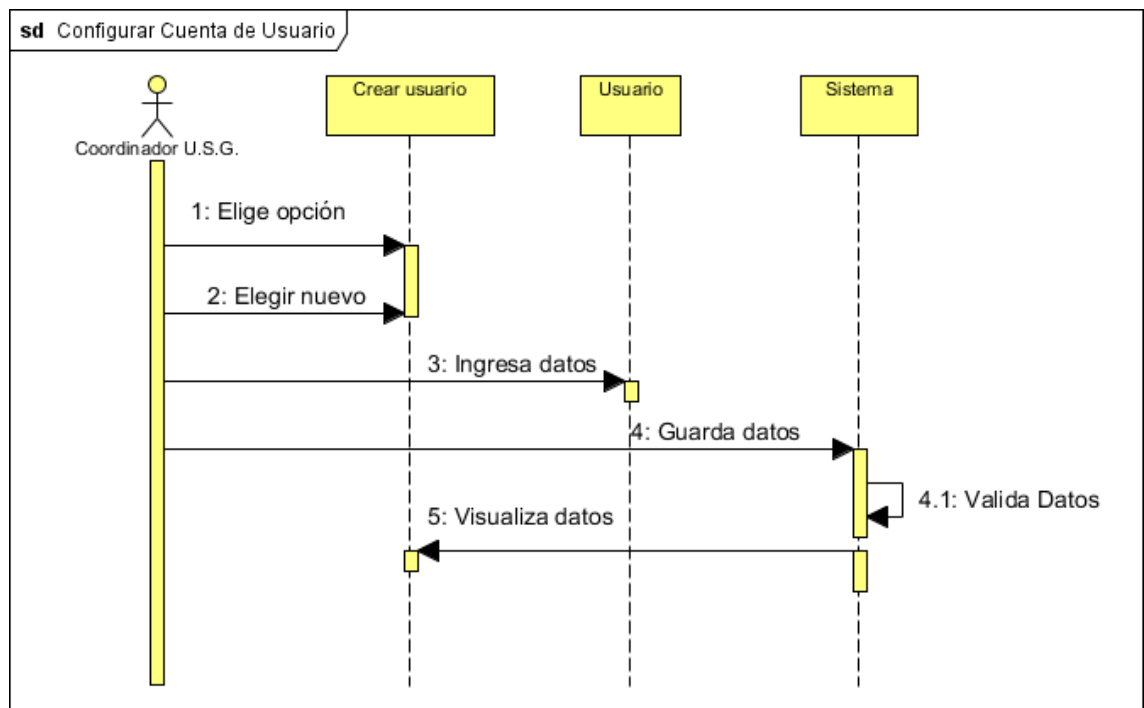


Figura 2. Configurar Cuenta de Usuario

4.5.3.3. Registra Personal de Servicio

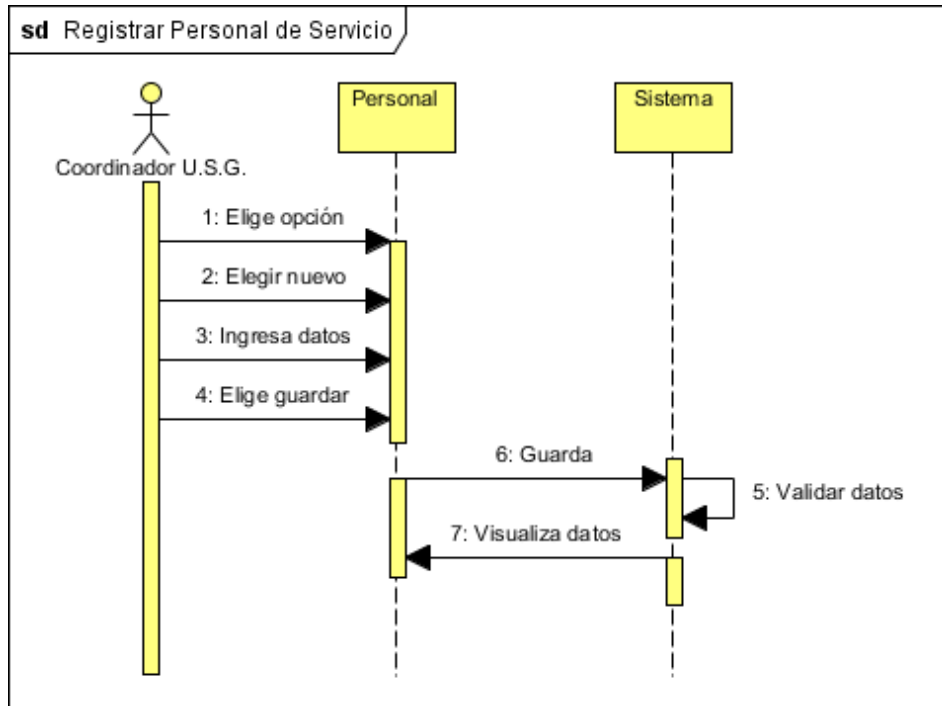


Figura 3. Registra Personal de Servicio

4.5.3.4. Ingreso de Actividades

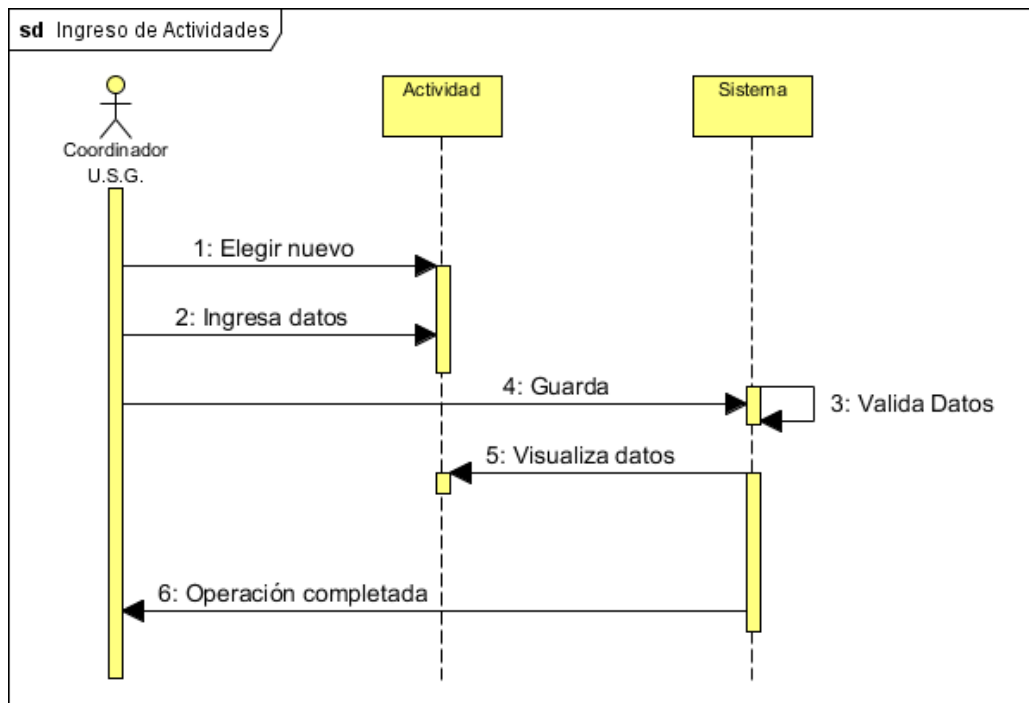


Figura 4. Ingreso de Actividades

4.5.3.5. Registra Ingreso de Herramientas

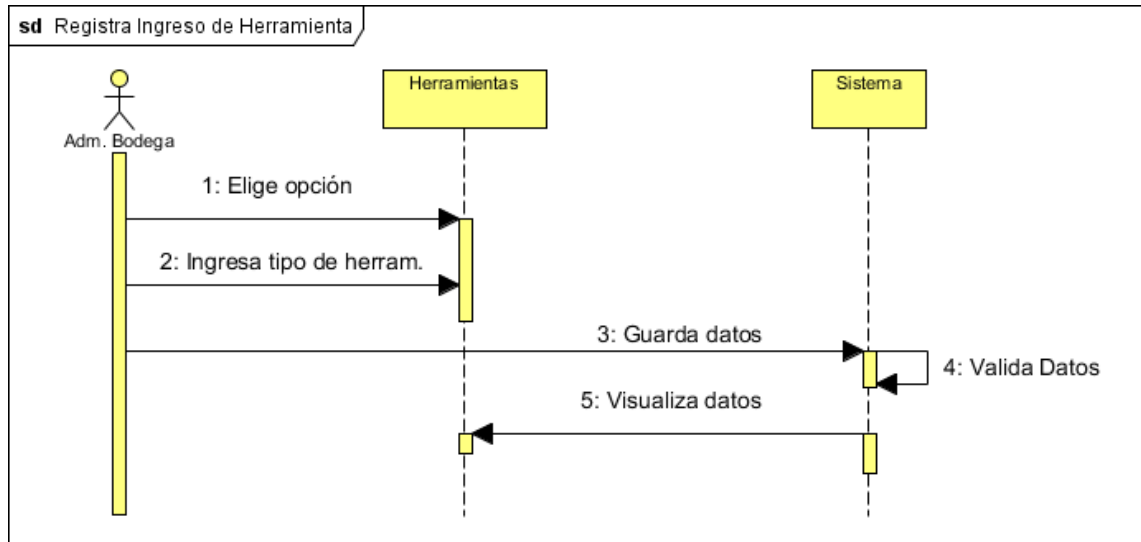


Figura 5. Registra Ingreso de Herramientas

4.5.3.6. Registra Ingreso de Material

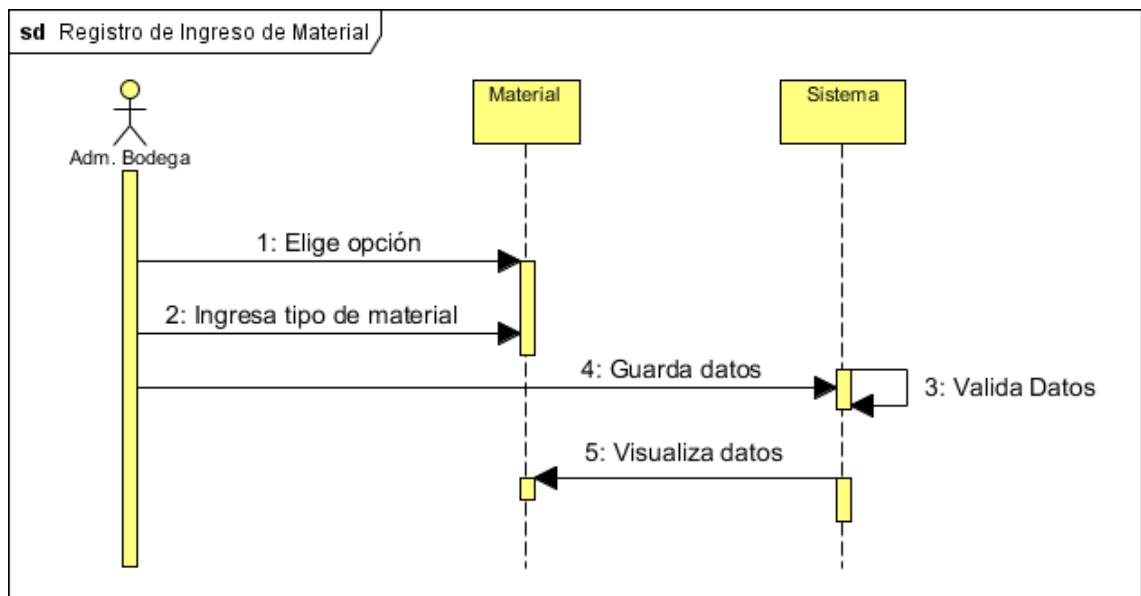


Figura 6. Registra Ingreso de Material

4.5.3.7. Registra Unidad de Medida

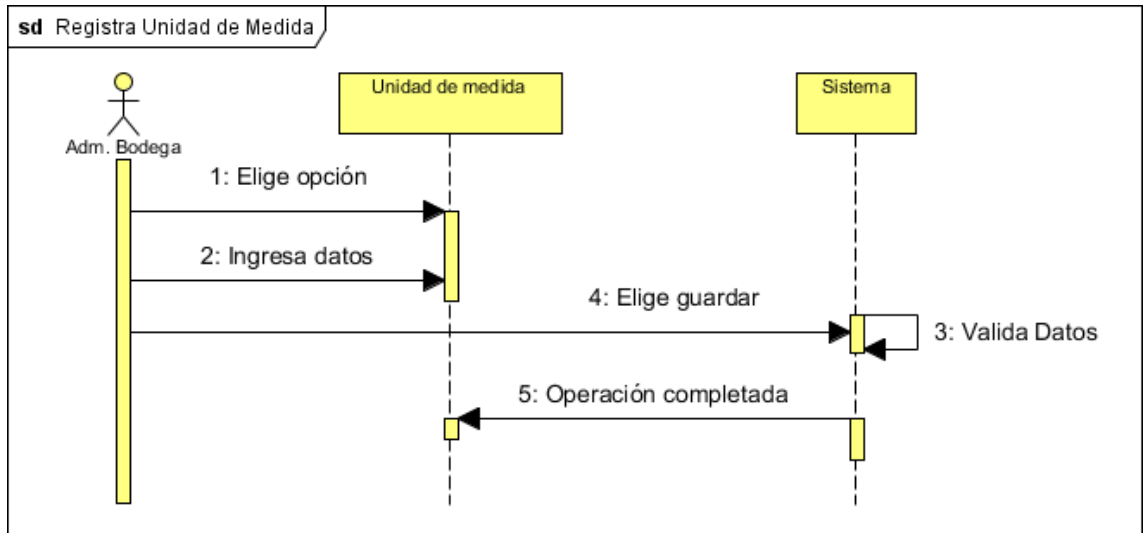


Figura 7. Registra Unidad de Medida

4.5.3.8. Registra Ingreso de Kardex

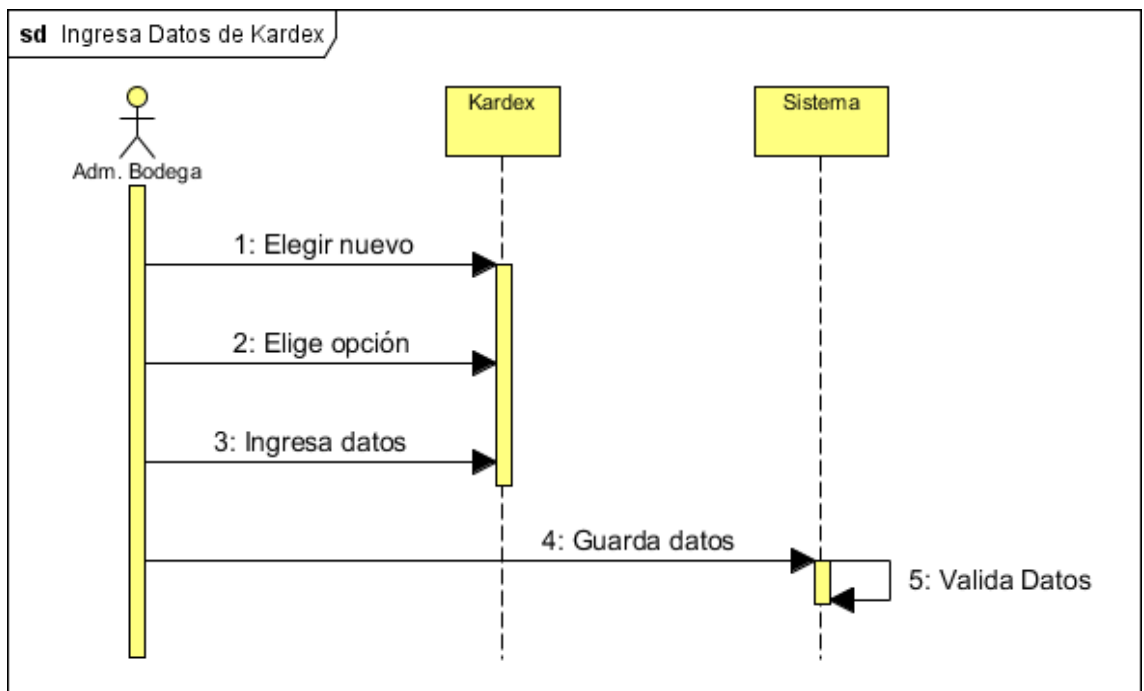


Figura 8. Registra Ingreso de Kardex

4.5.3.9. Realiza Consulta

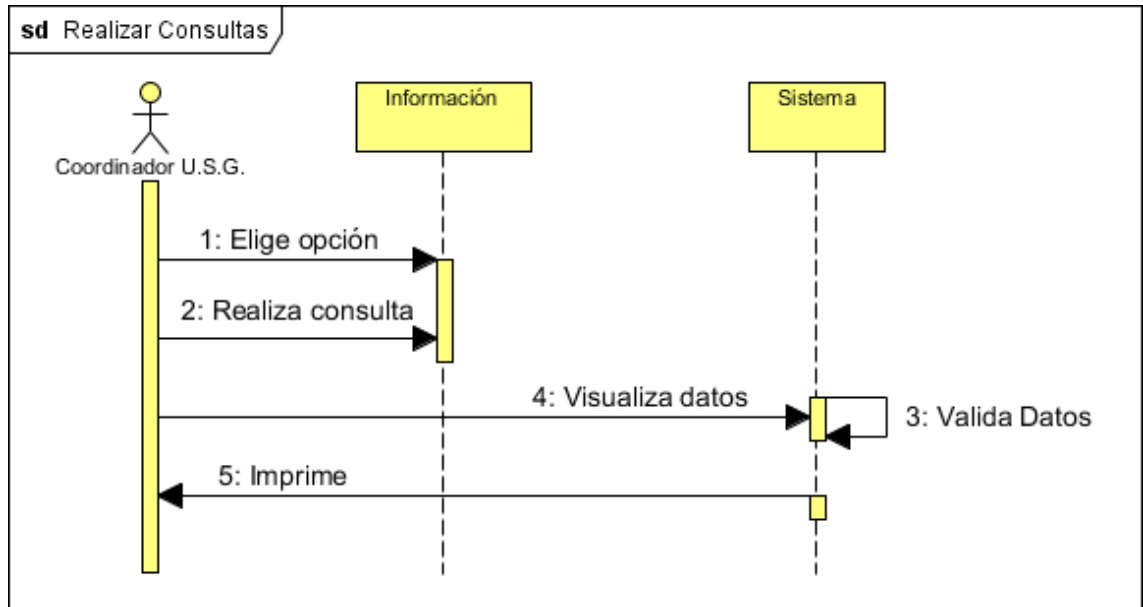


Figura 9. Realiza Consulta

4.5.3.10. Informe de Actividad

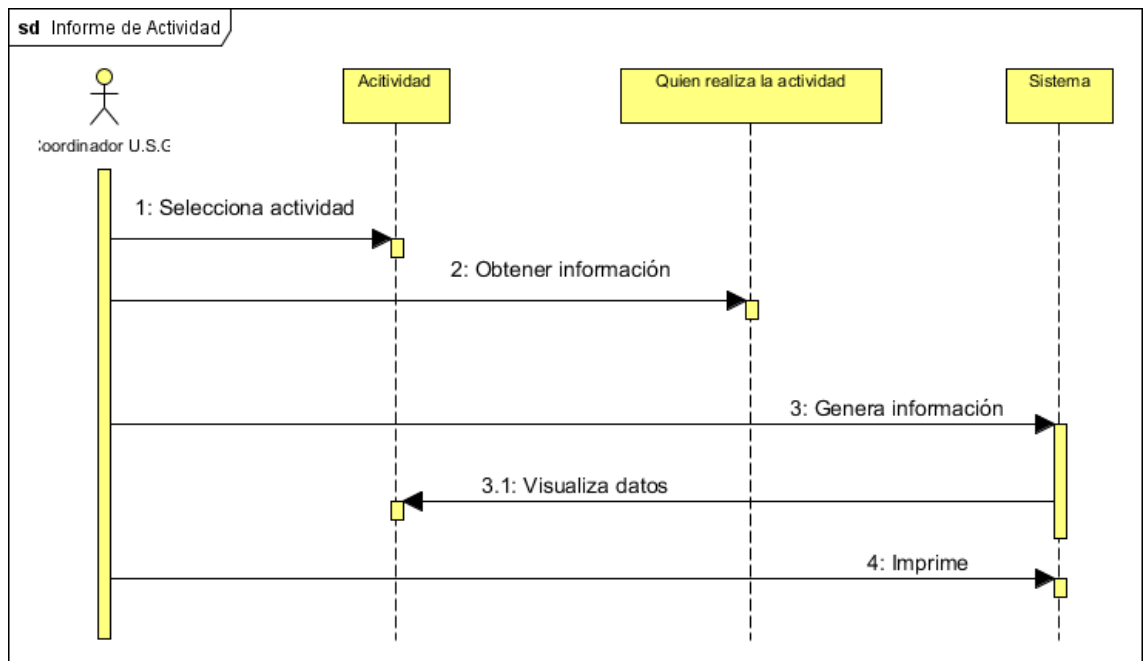


Figura 10. Informe de Actividad

4.5.3.11. Informe de Material

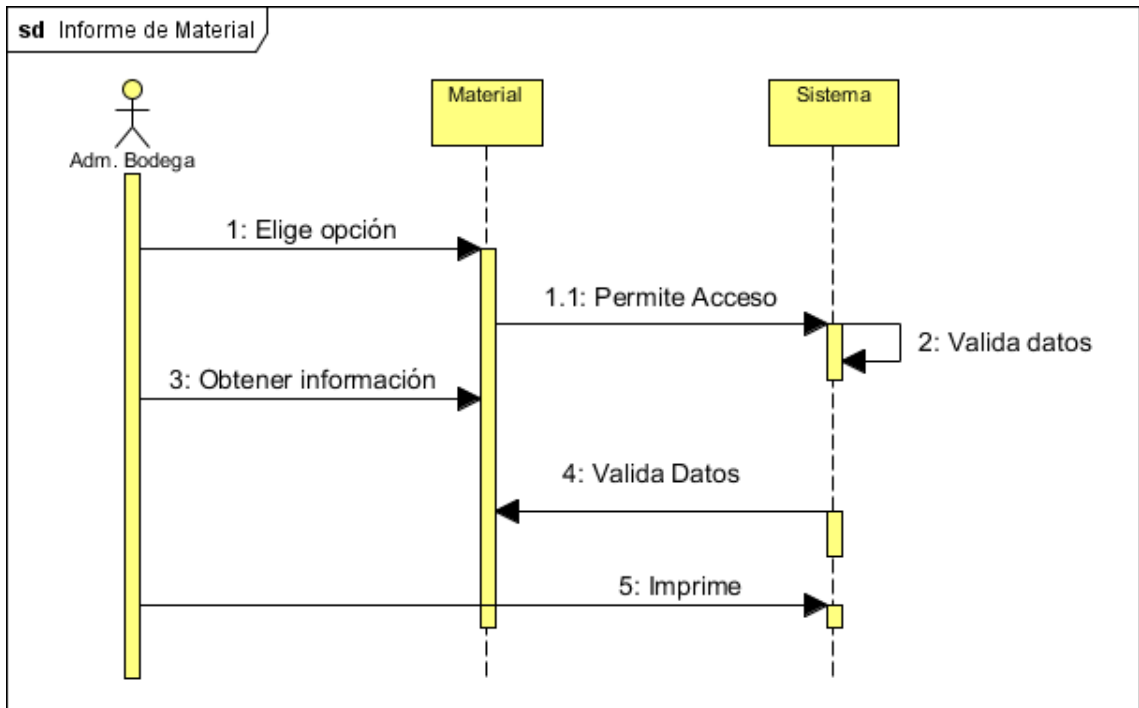


Figura 11. Informe de Material

4.5.3.12. Salida de Material

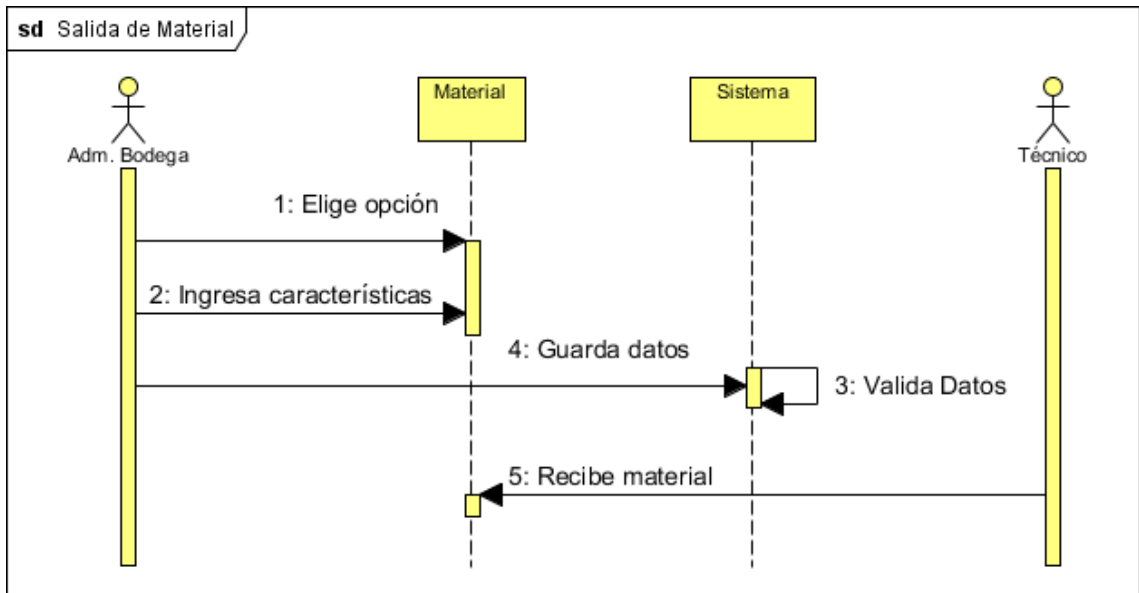


Figura 12. Salida de Material

4.5.3.13. Salida de Herramientas

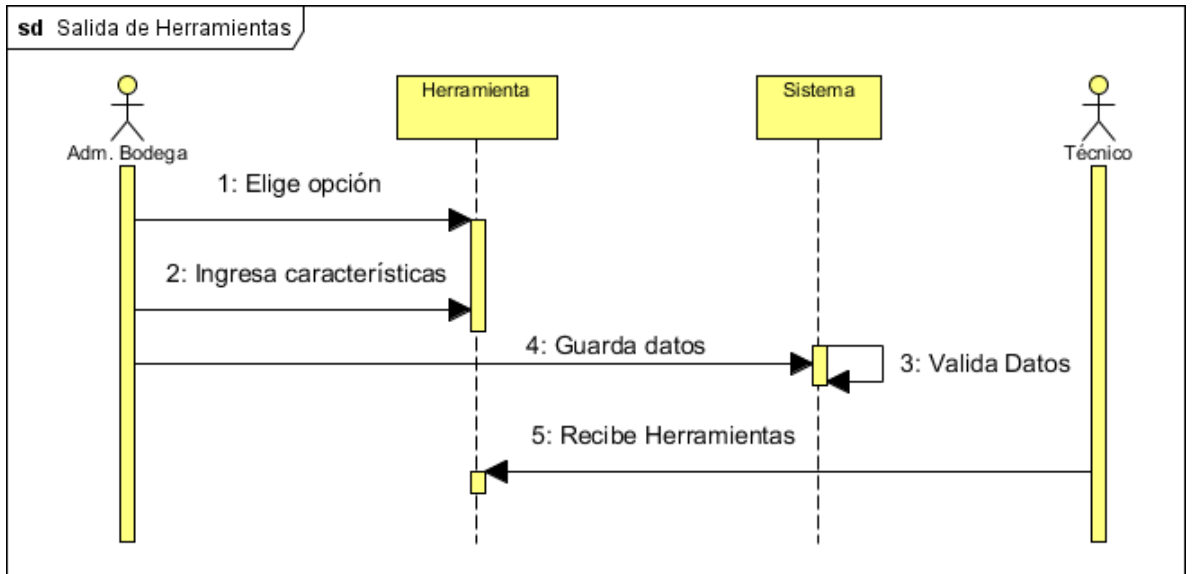
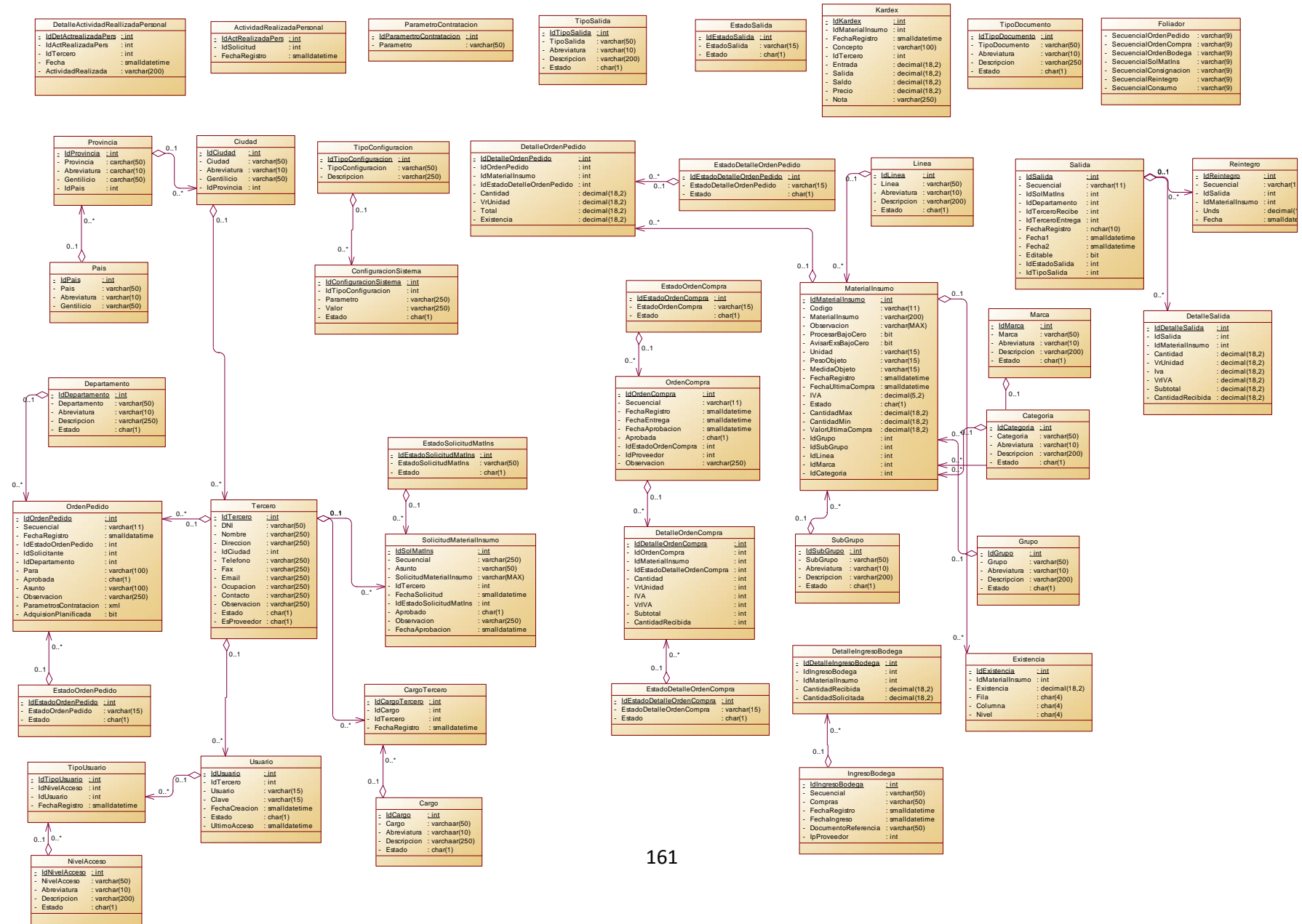


Figura 13. Salida de Herramientas

4.5.5. Diagrama de Clase



CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- Los flujos de informaciones que se llevan dentro de la empresa son desorganizados y deficientes, registrados en forma manual y archivada en carpetas lo cual origina la pérdida constante o confusión de información que podría a llegarse a superar con un sistema informático.
- Con los resultados del trabajo de campo se determinó que los flujos de información necesarios para el nuevo sistema son; registros de material, ingreso egreso del mismo, registro de labores, control de producción, reportes de control de producción, informe de material existente en stock, informe de material utilizado.
- El manejo de la información en la Unidad ha sido empírico, no ha existido un control adecuado, el archivo no cuenta con todos los movimientos detallados, al momento de recolectar la información el problema principal fue la desorganización y la poca información que se almacena, en hojas volantes, que son almacenadas en carpetas.
- Después de revisar como presta sus servicios la empresa se ha concluido que procesos deben constar en el sistema a implementar, para que ayuden a disminuir las tareas que actualmente se realizan generando costos innecesarios así se logrará maximizar los recursos que tiene la empresa.
- Ha sido necesario recurrir a diferentes autores para el desarrollo de esta investigación se ha podido sintetizar los diferentes criterios en el desarrollo del sistema, tomando en cuenta todos los aspectos como son diseño, programación, lenguaje, administración. Con la aplicación de esta información se ha diseñado el sistema automatizado a implementar en la empresa.
- Por el diseño de la base de Datos ha sido necesario un estudio profundo de las carencias con que cuenta Unidad de Servicios Generales de la UTEQ y

el tipo de sistema recomendado a implementar, al ser una dependencia pequeña no constituyó mayor utilización de recursos por lo cual se ha empleó el motor de Base de datos SQL Server 2008.

- Se empleó SQL Server 2008 por ser una herramienta eficiente en el procesamiento, almacenamiento, seguridad e integridad de los datos. Una de las características principales e importantes es la seguridad, si se desea copiar la base de datos del equipo para trasladarla a otra máquina es necesario desconectarla, además posee técnicas de respaldo y restauración que se pueden acceder desde el mismo sistema administrativo.
- Se utilizó Visual Basic.Net 2010 por ser un lenguaje de programación orientado a objetos, de fácil manejo y conexión con gestores de almacenamiento de datos, además es idóneo al momento de la creación de aplicaciones para Windows.

5.2. RECOMENDACIONES

- Los flujos de información en la empresa serán eficaz y eficientes si son organizados y procesados en un sistema que garanticen que los datos están disponibles en forma ordenada. Óptimos procesos de labores de campo ayudaran a mejorar el control en la fuerza laboral. Es recomendable un sistema informático porque mejorará la automatización de los procesos de información dentro de la empresa.
- Es necesaria una correcta utilización del sistema, seguir los procesos tal como se detalla en el manual de usuario para evitar trabas o problemas con el desarrollo del sistema en la empresa, se recomienda que utilicen el sistema personal que haya sido capacitado.
- Una vez implementado el sistema en la empresa se deberá llevar un mantenimiento y actualización constante, de existir algún cambio en las actividades de la empresa se informe al encargado para adaptar dichos cambios al sistema implementado.
- Se recomienda un manejo adecuado del sistema, para evitar información errónea, ya que entre los procesos existe una correlación y un colapso, lo que generaría pérdidas no deseadas a la empresa.
- De presentarse un crecimiento acelerado de la empresa recurrir a un ingeniero de software acorde a las nuevas necesidades que presente la empresa, para mantener la agilidad en la atención al cliente.
- Instalar el motor de SQL Server 2008 en la Máquina a ejecutar el sistema.
- Para el correcto manejo de la base de datos el sistema operativo más recomendable es el Windows 7, si desea ejecutar en Windows XP Service Pack 2 o superior debe tener actualizado el Framework, o ver en el manual de usuario los sistemas operativos compatibles con este programa.
Realizar respaldos (Backups) periódicamente y almacenarlos en medios extraíbles, el mismo que ayudará a restaurar la base de datos en caso de que suceda daños en el computador destino.

CAPÍTULO VI

BIBLIOGRAFÍA

6.1. BIBLIOGRAFÍA

Alfredo Weitzenfeld (2008). Ingeniería de Software Orientado a Objeto

Alfredo Weitzenfeld (2008). Ingeniería de Software Orientado a Objetos con UML. Java e Internet

Cesar Augusto Bernal (2006). Metodología de la investigación

Carlos E. Méndez A. (2006) Metodología Diseño y desarrollo del proceso de Investigación

Fernández Alarcón Vicente (2006), Desarrollo de sistemas de información, edición UPC, Pp 12-18.

Henry F. Korth. (2000).Fundamento de Base Datos

Solid Qualitylearning (2007). Base de Datos con SQL Server 2005

Roger Pressman (2007). Ingeniería de Software

6.2. Páginas Web

Álvarez Sara (2009).

<http://www.desarrolloweb.com/articulos/introduccion-base-datos.html>

Booch (2008)

[BOOCH Grady \(2008\) Lenguaje Unificado Modelado UML Primera edición Pp 35-48.](#)

Calderón Pacheco Betty (2009).

En <http://www.monografias.com/trabajos48/administracion->

educacion/administracion-educacion2.shtml

Charte Ojeda Francisco SQL SERVER, disponible en:
<http://www.casadellibro.com/libros/charteojedafrancisco/charte2ojeda32francisc>
[o](#)

Claudio Casares (2004).

<http://www.maestrosdelweb.com/editorial/tutsql1/>

ECLIPSE Process Framework Project (2008), Software Process EngineeringFramework disponible en:<http://epf.eclipse.org/wikis/openup>

Edgar Frank Codd, Científico informático inglés (23 de agosto de 1923 - 18 de abril de 2003), conocido por sus aportes a la teoría de bases de datos relacionales.

http://es.wikipedia.org/wiki/Base_de_datos_relacional

Kenneth, Carll. (2009). Historia de Sistemas, Disponible en:
<http://www.daedalus.es/inteligencia-de-negocio/sistemas-complejos/ingenieria-de-sistemas/disenio-de-sistemas/>

Marqués Andrés María Mercedes (2001-02-12) Ciencia de la Computación e Inteligencia Artificial. Bases de Datos (Universitat Jaume I OpenCourse) Metodología y Tecnología de la Programación - IS04 (Universitat Jaume I OpenCourse)

<http://www3.uji.es/~mmarques/f47/apun/node7.html>

Pérez Valdés Damián (2007).

<http://gestionemprededora.wordpress.com/2008/10/08/%C2%BFplanillas-de-calculo-o-bases-de-datos/>

Pinto molina María (2009).

http://www.mariapinto.es/e-coms/bases_datos.htm#bd1
<http://www.mariapinto.es/web/mainframe.htm>

Raúl Uranga Cruz(2008).

<http://www.monografias.com/trabajos12/basdat/basdat.shtml>

Rodolfo Quispe-Otazu (2007).

<http://www.rodolfoquispe.org/blog/que-es-la-ingenieria-de-software.php>

Ruiz Calleja (2009)

<http://www.monografias.com/trabajos38/direccion-gestion-educativa/direccion-gestion-educativa.shtml#una>

Agile Manifiesto. (2009). Manifiesto for Agile Software Development
<http://agilemanifesto.org> Kent Beck, James Grenning, Robert C. Martin, et. al.

James Rumbaugh. (2010). Modelado y diseño orientados a objetos
Metodología OMT. Editorial Prentice Hall Primerareimpresión.

Markus Rerych (2007). Wasserfallmodell>Entstehungskontext,
InstitutfürGestaltungs- und Wirkungsforschung, TU-Wien.

Peña (2006).

<http://www.econlink.com.ar/sistemas-informacion/definicion>

Peralta (2008). (2008).

<http://www.econlink.com.ar/sistemas-informacion/definicion>

Manuel Peralta (2008).

<http://www.monografias.com/trabajos7/sisinf/sisinf.shtml>

Econ. Manuel A. Pulgar

<http://www.gestiopolis.com/dirqp/adm/produccion.htm>

Paolo castillo Rubio (2007)

http://www.slideshare.net/ptah_enki/sistemas-de-control

CAPÍTULO VII

ANEXOS

7.1. Cuestionario de Encuestas y Entrevistas

Encuesta para los usuarios que reciben los servicios de la Unidad de Servicios Generales de la UTEQ.

1. **Cómo califica usted la atención que brinda la Unidad de Servicios Generales.**

Regular: Bueno: Muy Bueno: Excelente:

2. **¿Cree usted que el trabajo de la U.S.G. se realiza de manera técnica y profesional?**

Si: No:

En caso de que la respuesta sea negativa indique él por qué

3. **Cree usted que la U.S.G realiza su trabajo de manera eficaz eficiente y oportuna?**

Si: No:

En caso de que la respuesta sea negativa indique él por qué

4. **¿Está de acuerdo que en la Unidad de Servicios Generales se automatice con un sistema informático?**

Si No

UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
UNIDAD DE ESTUDIOS A DISTANCIA

Encuesta para el personal de la Unidad de Servicios Generales los cuales están conformados por (Electricidad, albañilería, gasfitería, pintura, climatización, jardinería)

1. Como califica la atención que brinda la Unidad de Servicios Generales a los Usuarios de la UTEQ.

Regular: Bueno: Muy Bueno: Excelente:

2. ¿Tiene problemas al realizar el pedido de los materiales para realizar su trabajo?

Si No

Si la respuesta es afirmativa, describa el tipo de problema

.....
.....

3. Usted emite un informe de daños cuando se le asigna un trabajo?

Si No

Si la respuesta es afirmativa, describa como lo realiza

.....
.....

4. ¿Está de acuerdo que en la Unidad de Servicios Generales se automatice con un sistema informático?

Si No

Si la respuesta es afirmativa o negativa, determine el porqué.

.....
.....

UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
UNIDAD DE ESTUDIOS A DISTANCIA

Entrevista para el Coordinador de la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo.

8. ¿A qué se dedica la Unidad de Servicios Generales?

9. ¿Cómo está organizada administrativamente la unidad de Servicios Generales?

10. ¿Describa los procesos que se ejecutan en la Unidad de Servicios Generales?

11. ¿Cómo realiza el control de stock de productos, materiales confiable en su unidad?

12. ¿Cree usted que al contar con un sistema automatizado, el proceso de los registros, elaboración de informes y reportes, le permitirá optimizar su tiempo de trabajo y podrá realizar más rápida la entrega de los mismos?

Si No

Por qué?

.....

.....

UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
UNIDAD DE ESTUDIOS A DISTANCIA

Entrevista al administrador de Bodega de la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo.

6. ¿Para registrar el ingreso/salida de los materiales, bienes y suministros, lo realiza de manera manual o automatizada?

Manual Automatizada

Describa como realiza este proceso

.....
.....

7. Cuáles son los documentos que utiliza en la ejecución de los procesos en la U.S.G.

.....
.....

8. ¿Considera usted que con un sistema automatizado podría obtener resultados precisos e inventario real de los materiales existentes en bodega?.

Si No

Por qué?

.....
.....

9. ¿Cree usted que al contar con un sistema automatizado, el proceso de los registros, elaboración de informes y reportes, le permitirá optimizar su tiempo de trabajo y podrá realizar más rápida la entrega de los mismos?

Si No

Por qué?

.....
.....

1. **¿Para registrar el ingreso/salida; entrega/recepción de los materiales, bienes y suministros, lo realiza de manera manual o automatizada?**

Manual

Automatizada

2. **¿Considera usted que con un sistema automatizado podría obtener mejores resultados, más precisos e inventario real de los materiales existente en bodega?**

Si

No

3. **¿Cree usted que al contar con un sistema automatizado, el proceso de los registros, elaboración de informes y reportes, le permitirá optimizar su tiempo de trabajo y podrá realizar más rápida la entrega de los mismos?**

Si

No

4. **Determine los documentos que utiliza en la ejecución de los procesos en la U.S.G.**

Los documentos que se utilizan en la ejecución de los procesos en la Unidad de Servicios Generales son:

Comprobante de egreso e ingreso de material, registro de material, Entrega de material, informe de material en stock, informe de ingreso de material, informe de entrega de material etc.

7.2. Manual de Usuario

7.2.1. Introducción

El presente manual es facilitar al usuario la operación de las distintas pantallas de captura que tendrán acceso al “SISTEMA INFORMÁTICO PARA LA GESTIÓN Y EL CONTROL DE SERVICIOS GENERALES DE LA UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO”. Este Sistema de Información fue creado con el propósito primordial de llevar una mejor organización de toda la información de la Unidad de Servicios Generales.

7.2.2. Objetivos del Manual

El objetivo del manual es describir el funcionamiento del Sistema, mediante gráficos y descripción detallada.

7.2.3. Especificaciones Técnicas

Para implementar el Sistema Informático para la Gestión y el Control de Servicios Generales de la Universidad Técnica Estatal de Quevedo se requirió lo siguiente:

Cuadro 62. Especificaciones Técnicas

Hardware	Software
Pc Intel Core 2 Duo	Sistemas Operativos compatibles: XP, Windows 7.
Memoria de 2gb recomendado	Framework 3.5,
Impresora	Reportview 8.0
	SQL Server 2008

Elaborado por. El Autor.

7.2.4. Funcionamiento del Sistema

Inicio de Sesión

Para acceder al Sistema Informático para la Gestión y el Control de Servicios Generales de la Universidad Técnica Estatal de Quevedo, usted debe realizar doble clic sobre el icono que se encuentra en el escritorio, aparecerá la pantalla de inicio al sistema, a continuación se visualizará un formulario que solicitará el ingreso de Usuario y Contraseña. (Ver Figura 1)

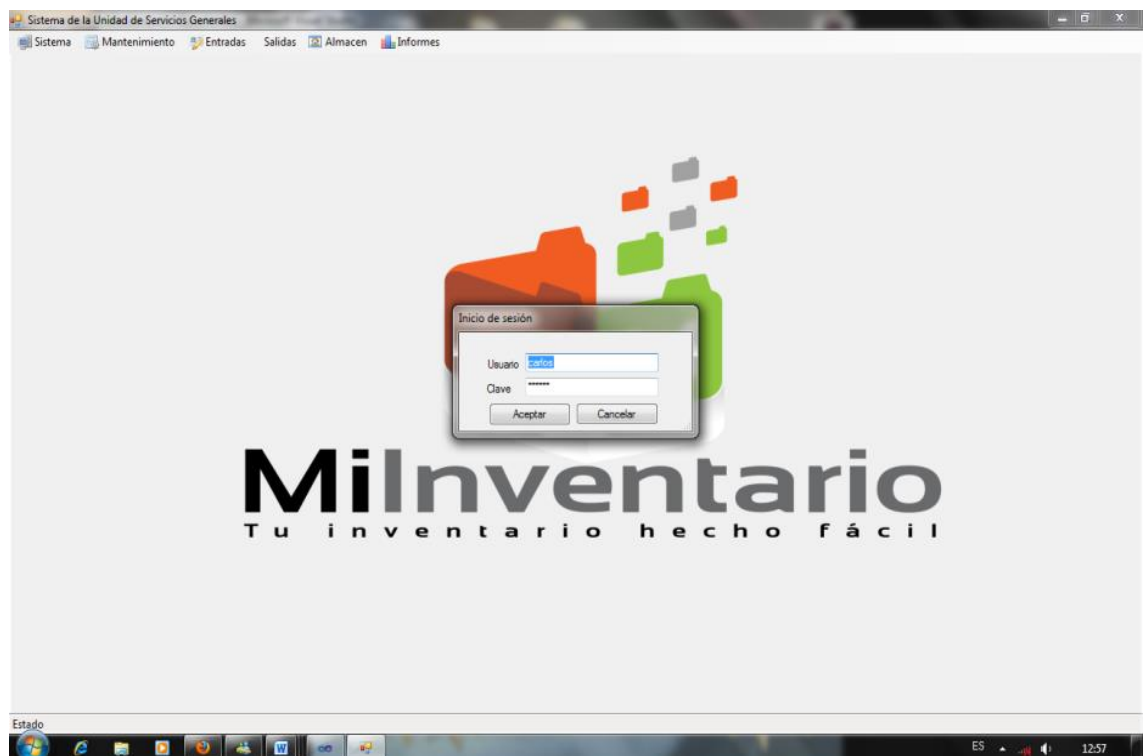


Figura 1.

Al ingresar el usuario y clave correcta se visualizará el menú de las opciones del sistema para ser manipulado por el administrador o a quien se le haya asignado una clave al sistema (Ver figura 2).

- **Menú Sistema**

Está compuesta por las siguientes opciones.

- 1.- Usuarios
- 2.- Cambiar de Usuarios
- 3.- Configuración del Sistema
- 4.- Serrar Sesión

Se podrá elegir cualquiera de estas opciones que se presenta en la siguiente pantalla.



Figura 2

Al dar un clic en la opción de **Sistema** se despliega un menú para ingresar se presiona clic en **Usuario**, en la que se mostrará la pantalla **Nuevo Usuario**, en ella dar clic en nuevo la cual permite ingresar un nuevo usuario, cuenta con las opciones Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 3).

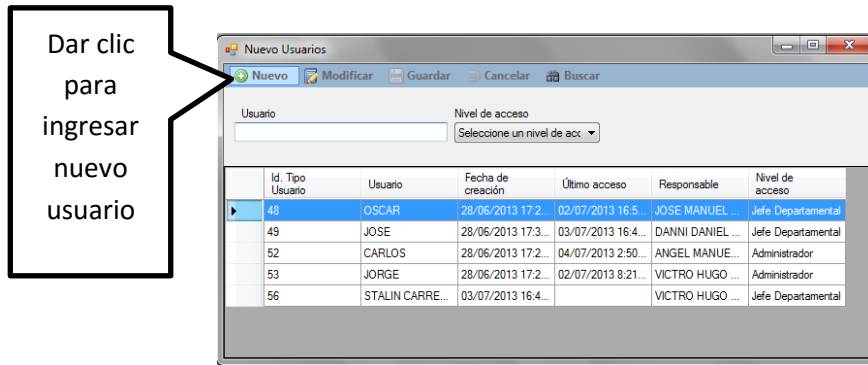


Figura 3

Se procede a ingresar el nombre de usuario, clave y quien se responsabiliza de su ingreso, clic en la opción **Estad** se despliega un menú y dar clic en **habilitar**, seleccionamos un **Nivel de acceso** en el que escoge **Bodeguero**, **Jefe Departamental** y **Jefe de Bodega** y dar clic en **Agregar** y guardar listo. (Ver Figura 4).

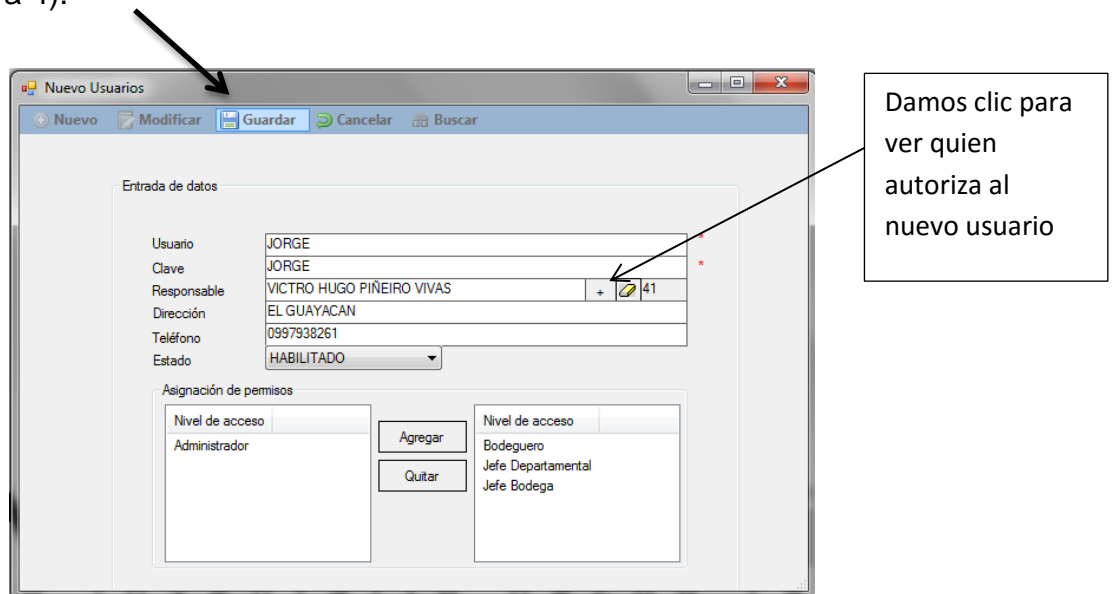


Figura 4

Para modificar un usuario seleccionar la lista, (Ver Figura 5) dar clip en la opción Modificar, se muestran los datos a modificar en el formulario se hacen las modificaciones y presionar clip en botón Guardar. (Ver Figura 6).

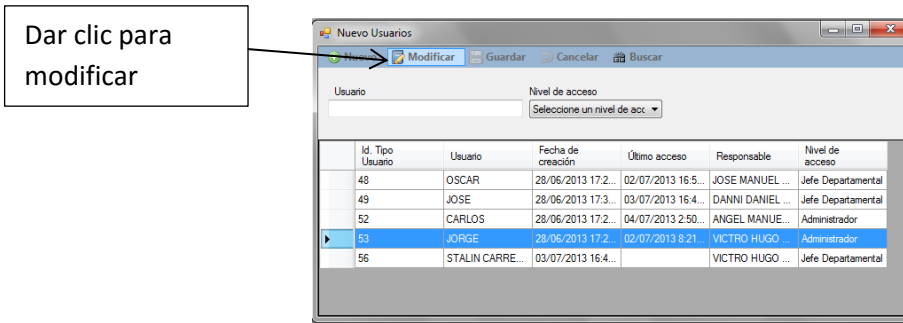


Figura 5

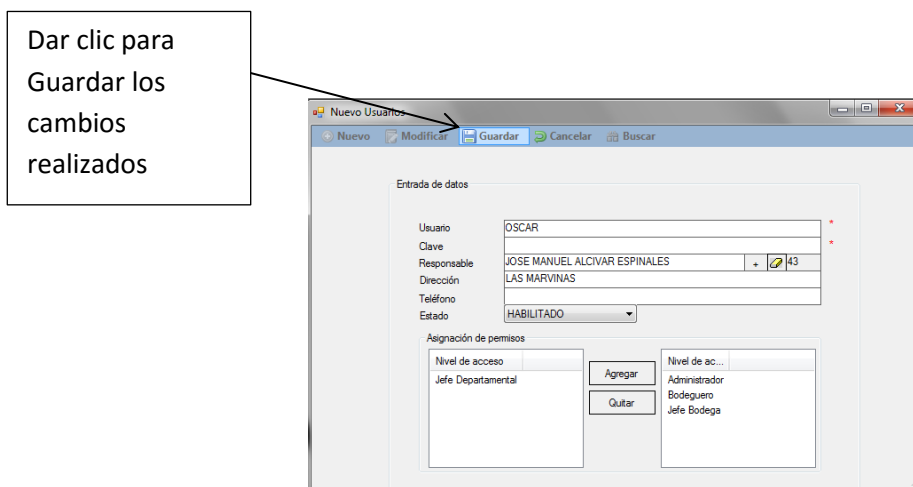
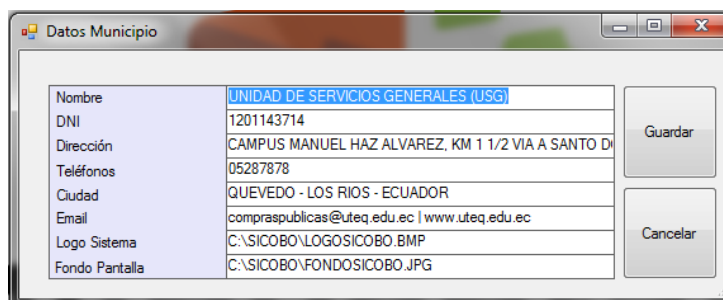


Figura 6.

Para Configurar el Sistema

Dar clic en **Sistemas** en el cual se despliega un menú, selecciono la opción, dando clic en **Configurar Sistema**, se muestra el formulario y realizar cambios correspondientes y guardar.

Nota esta acción solo lo realiza el administrador (Ver Figura 7).



Nombre	UNIDAD DE SERVICIOS GENERALES (USG)
DNI	1201143714
Dirección	CAMPUS MANUEL HAZ ALVAREZ, KM 1 1/2 VIA A SANTO D
Teléfonos	05287878
Ciudad	QUEVEDO - LOS RIOS - ECUADOR
Email	compraspublicas@uteq.edu.ec www.uteq.edu.ec
Logo Sistema	C:\SICOBO\LOGOSICOBO.BMP
Fondo Pantalla	C:\SICOBO\FONDOSICOBO.JPG

Figura 7

- **Menú Mantenimiento**

En el menú de **Mantenimiento** del sistema se presentan varias opciones para el uso correspondiente del usuario, que son:

- 1.- País
- 2.- provincia
- 3.- Ciudad
- 4.- Grupo
- 5.- Sub Grupo
- 6.- Marca
- 7.- Línea
- 8.- Departamentos
- 9.- Personas / Entidades
- 10.- Cargos

11.- Asignación de cargo

(Ver Figura 8).



Figura 8

Al seleccionar la opción **País** se presenta un formulario en el que visualizamos los países ingresados, cuenta con las opciones de Nuevo, Modificar, Guardar, Cancelar, buscar (Ver Figura 9).

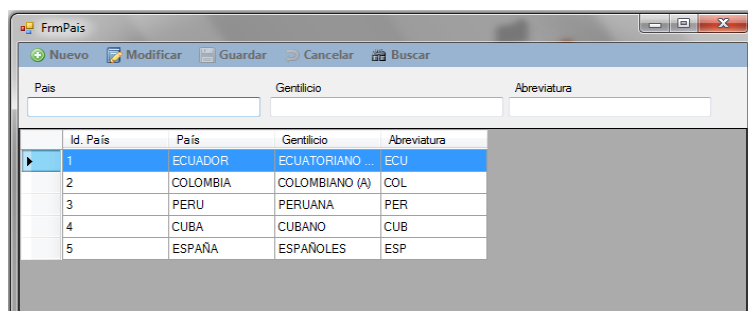


Figura 9

Si se desea ingresar un nuevo país dar clic en la opción **Nuevo** ingresar el nombre del país el gentilicio y abreviatura ingresar los datos clic en guardar (Ver Figura 10).

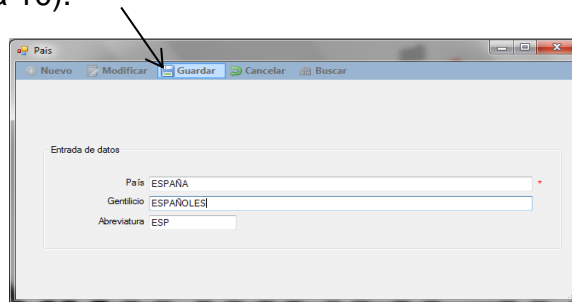


Figura 10

Si se desea modificar los datos se selecciona la información dar clic en modificar, (Ver Figura 11).

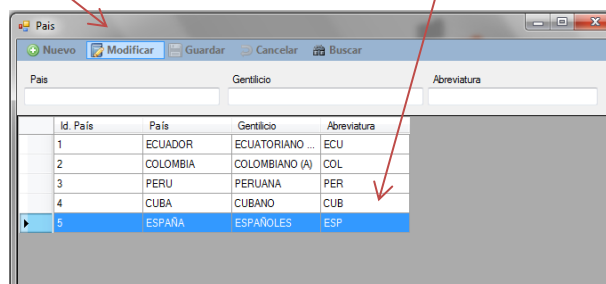


Figura 11

Aparece otra ventana en la que se cambia o modifica, dar clic en la opción guardar, y listo (Ver Figura 12).

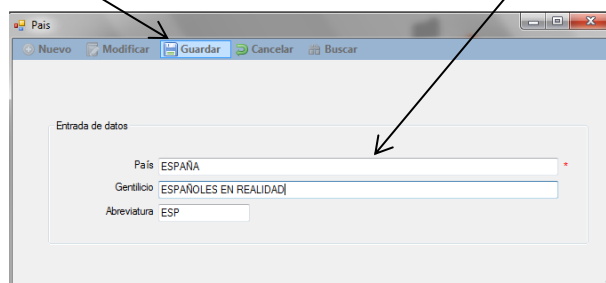


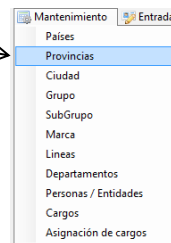
Figura 12

Con los nuevos cambios nos quedaría la información de esta forma (Ver Figura 13).

Id. Pais	Pais	Gentilicio	Abreviatura
1	ECUADOR	ECUATORIANO(A)	ECU
2	COLOMBIA	COLOMBIANO (A)	COL
3	PERU	PERUANA	PER
4	CUBA	CUBANO	CUB
5	ESPAÑA	ESPAÑOLES EN REALIDAD	ESP

Figura 13

Al seleccionar la opción **Provincias** en el que visualiza las provincias cuenta con las opciones de Nuevo, Cancelar y Buscar. (Ver Figura 14).



se presenta un formulario ingresadas, y además Modificar, Guardar,

IdProvincia	Provincia	Gentilicio	Abreviatura	IdPais	Pais
1	LOS RIOS	RIOSENSE	L-R	1	ECUADOR
2	GUAYAS	GUAYAQUILEÑO	GYE	1	ECUADOR
3	PICHINCHA	PICHINCHANO	PICH	1	ECUADOR
5	AZUAY	AZUAYO	AZ	1	ECUADOR
6	BOLIVAR	BOLIVIANO	BOLI	1	ECUADOR
7	CAÑAR	CAÑALERO	CÑ	1	ECUADOR
8	CARCHI	CARCHINCE	CARCHI	1	ECUADOR
9	CHIMBORAZO	CHIMBORANSE...	CHIM	1	ECUADOR
10	COTOPAXI	COTOPAXENCE	COTO	1	ECUADOR
11	EL ORO	ORENCE	ORO	1	ECUADOR
12	ESMERALDAS	ESMERALDEÑO	ESMER	1	ECUADOR
13	GALAPAGOS	GALOPENCE	GALA	1	ECUADOR

Figura 14

Si se desea ingresar otra provincia dar clic en la opción **Nuevo** ingresa el nombre de la provincia el gentilicio, abreviatura y selecciona

el país al que pertenece y datos clic en guardar (Ver Figura 15).

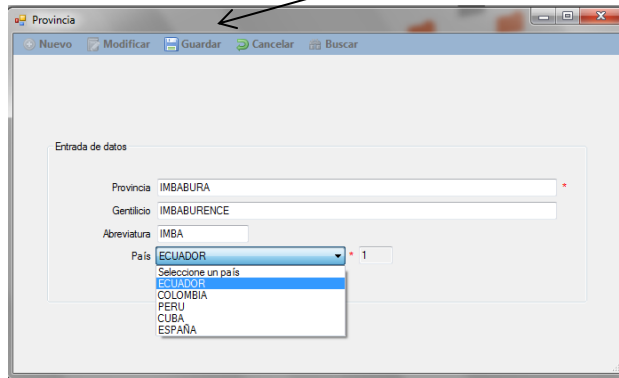


Figura 15

Si se desea modificar los datos se selecciona la información dar clic en modificar, (Ver Figura 16).

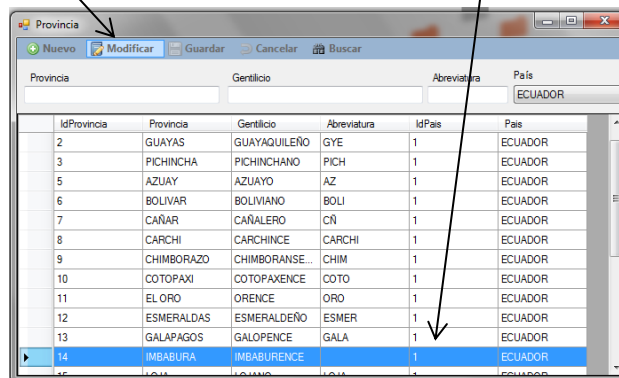


Figura 16

Aparece otra ventana en la cambiar o modificar, dar clic en la opción guardar, y listo. (Ver Figura 17).

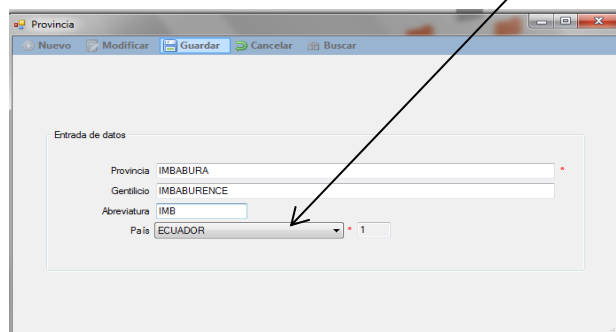


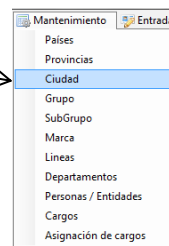
Figura 17

Con los nuevos cambios nos quedaría la información de esta forma (Ver Figura 18).

IdProvincia	Provincia	Gentilicio	Abreviatura	IdPais	Pais
2	GUAYAS	GUAYAQUILEÑO	GYE	1	ECUADOR
3	PICHINCHA	PICHINCHANO	PICH	1	ECUADOR
5	AZUAY	AZUAYO	AZ	1	ECUADOR
6	BOLIVAR	BOLIVIANO	BOLI	1	ECUADOR
7	CAÑAR	CAÑALERO	CAÑ	1	ECUADOR
8	CARCHI	CARCHINCE	CARCHI	1	ECUADOR
9	CHIMBORAZO	CHIMBORANSE...	CHI	1	ECUADOR
10	COTOPAXI	COTOPAXENCE	COTO	1	ECUADOR
11	EL ORO	ORENCE	ORO	1	ECUADOR
12	ESMERALDAS	ESMERALDEÑO	ESMER	1	ECUADOR
13	GALAPAGOS	GALOPENCE	GALA	1	ECUADOR
14	IMBABURA	IMBABURENCE	IMB	1	ECUADOR

Figura 18

Al seleccionar la opción **Ciudad** se el que visualizar las ciudades cuenta con las opciones de Nuevo, Cancelar, buscar (Ver Figura 19).



presenta un formulario en ingresadas, y además Modificar, Guardar,

Id.Ciudad	Ciudad	Gentilicio	Abreviatura	Pais	Provincia
1	CUENCA	CUENCANO	CUEN	ECUADOR	LOS RIOS
2	QUEVEDO	QUEVEDEÑO	QVDO	ECUADOR	LOS RIOS
3	MACHALA	MACHALEÑO	MAC	ECUADOR	LOS RIOS
4	GUAYAQUIL	GUAYAQUILEÑO	GYE	ECUADOR	LOS RIOS
5	QUITO	QUITEÑO	QUI	ECUADOR	LOS RIOS
6	AMBATO	AMBATEÑO	AMB	ECUADOR	LOS RIOS
7	ESMERALDAS	ESMERALDEÑO	ESME	ECUADOR	LOS RIOS
8	SANTO DOMINGO	SANTO DOMINGO	SAT.DOM	ECUADOR	LOS RIOS
9	LOJA	LOJA	LOJA	ECUADOR	LOS RIOS
10	PUYO	PUYANO	PUY	ECUADOR	LOS RIOS

Figura 19

Si se desea ingresar otra ciudad dar clic en la opción **Nuevo** ingresar el nombre de la ciudad el gentilicio, abreviatura dar clic donde

dice país, se despliega un menú, seleccionar el país a la que pertenece, de igual forma dar clic en provincia, se despliega un menú, seleccionar la provincia al que pertenece y clic en guardar (Ver Figura 20).

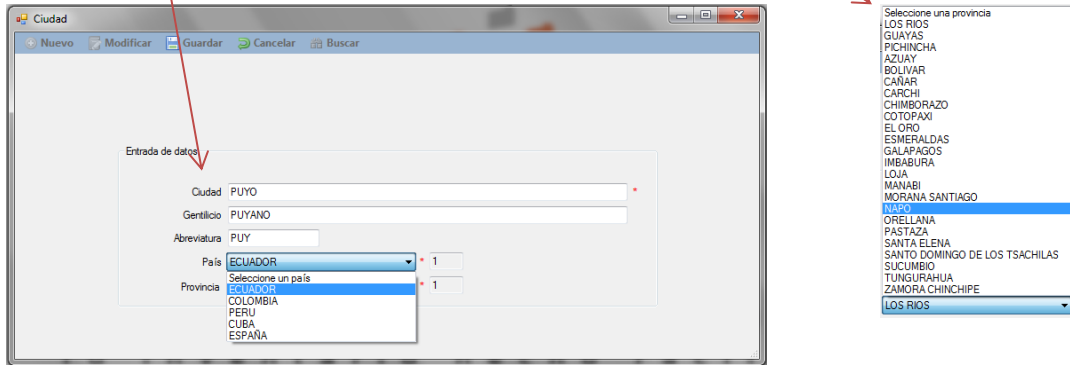


Figura 20

Si se desea modificar los datos seleccionar la información dar clic en modificar, (Ver Figura 21).

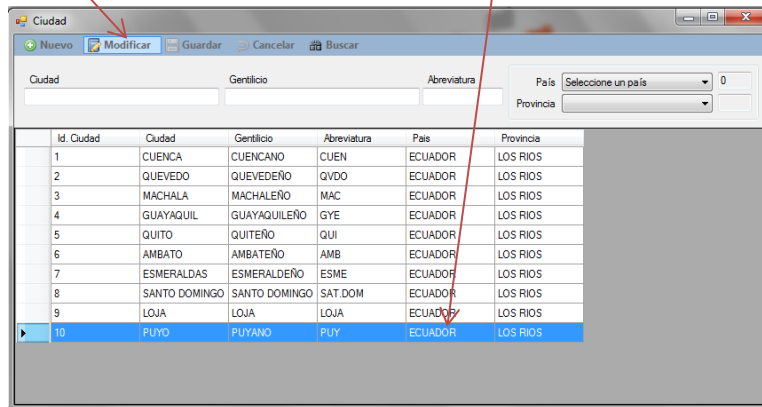


Figura 21

Aparece otra ventana en la cambiar o modificar, dar clic en la opción guardar, y listo (Ver Figura 22).

The screenshot shows a window titled 'Ciudad' with a menu bar containing 'Nuevo', 'Modificar', 'Guardar', 'Cancelar', and 'Buscar'. The 'Guardar' button is highlighted. Below the menu is a form titled 'Entrada de datos' with the following fields: 'Ciudad' (text input with 'PUYO'), 'Gentilicio' (text input with 'PUYATANO'), 'Abreviatura' (text input with 'PUY'), 'País' (dropdown menu with 'ECUADOR'), and 'Provincia' (dropdown menu with 'LOS RIOS').

Figura 22

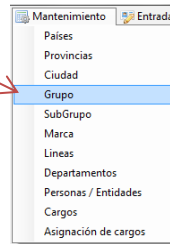
Con los nuevos cambios nos quedaría la información de esta forma (Ver Figura 23).

The screenshot shows a window titled 'Ciudad' with a menu bar containing 'Nuevo', 'Modificar', 'Guardar', 'Cancelar', and 'Buscar'. Below the menu are fields for 'Ciudad', 'Gentilicio', 'Abreviatura', 'País' (dropdown menu with 'ECUADOR'), and 'Provincia' (dropdown menu with 'Seleccione una provincia'). Below these fields is a table with the following data:

Id. Ciudad	Ciudad	Gentilicio	Abreviatura	País	Provincia
1	CUENCA	CUENCANO	CUEN	ECUADOR	LOS RIOS
2	QUEVEDO	QUEVEDENO	QVDO	ECUADOR	LOS RIOS
3	MACHALA	MACHALERO	MAC	ECUADOR	LOS RIOS
4	GUAYAQUIL	GUAYAQUILEÑO	GYE	ECUADOR	LOS RIOS
5	QUITO	QUITEÑO	QUI	ECUADOR	LOS RIOS
6	AMBATO	AMBATENO	AMB	ECUADOR	LOS RIOS
7	ESMERALDAS	ESMERALDEÑO	ESME	ECUADOR	LOS RIOS
8	SANTO DOMINGO	SANTO DOMINGO	SAT.DOM	ECUADOR	LOS RIOS
9	LOJA	LOJA	LOJA	ECUADOR	LOS RIOS
10	PUYO	PUYATANO	PUY	ECUADOR	LOS RIOS

Figura 23

Al seleccionar la opción **Grupo** se el que visualizamos el grupo cuenta con las opciones de Nuevo, Cancelar y Buscar. (Ver Figura 24).



presenta un formulario en ingresado, y además Modificar, Guardar,

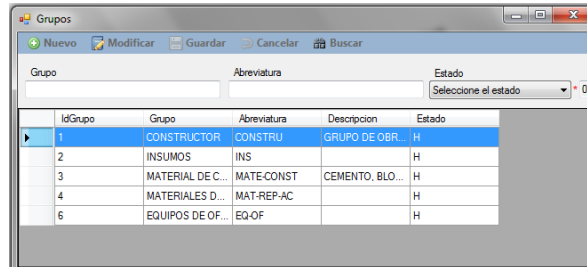


Figura 24

Si se desea ingresar otra ciudad dar clic en la opción **Nuevo** ingresar el nombre del grupo de producto le damos una abreviatura si se desea se le da una descripción o se puede realizarlo otro día, en la opción estado dar clic en habilitar si selecciona deshabilitar quiere decir que se guarda la información pero no se la puede utilizar aun, una vez realizados todos los procesos se da clic en guardar (Ver Figura 25).

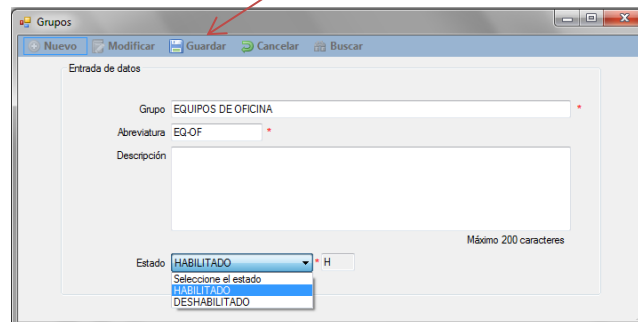


Figura 25

Si se desea modificar los datos seleccionar la información damos clic en modificar, (Ver Figura 26).

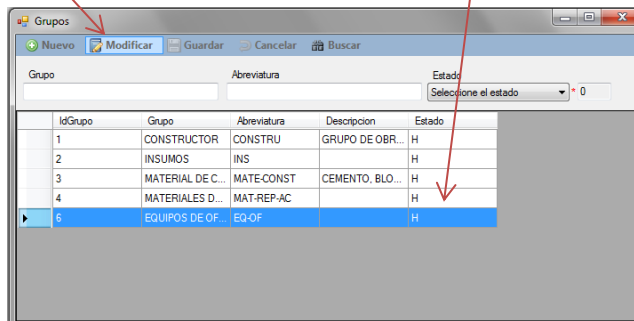


Figura 26

Aparece otra ventana en la cambiar o modificar, damos clic en la opción guardar, y listo (Ver Figura 27).

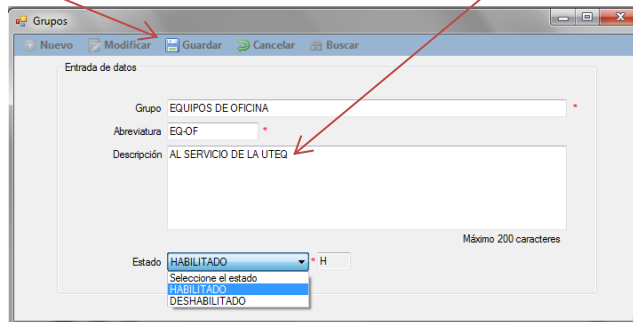


Figura 27

Con los nuevos cambios nos quedaría la información de esta forma (Ver Figura 28).

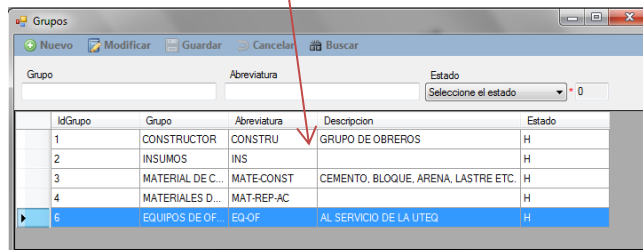
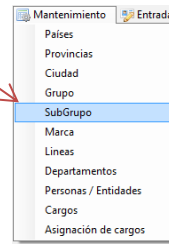


Figura 28

Al seleccionar la opción **Sub Grupo** en el que visualizar los sub grupo cuenta con las opciones de Nuevo, Cancelar y Buscar. (Ver Figura 29).



se presenta un formulario ingresadas, y además Modificar, Guardar,

IdSubGrupo	SubGrupo	Abreviatura	Descripcion	Estado
1	ALBAÑIL	ALB	OBRERO	H
2	CARPINTERO	CAR	OBRERO	H
3	CLIMATIZACION	CLIM	OBRERO	H
4	PINTOR	PINT	PERSONAJE GU...	H
5	JARDINERO	JARD	OBRERO	H
6	PLOMERO	PLOM	OBRERO	H

Figura 29

Si se desea ingresar otro sub grupo damos clic en la opción **Nuevo** ingresamos el nombre del sub grupo le abreviar si se desea se le da una descripción o se puede realizarlo otro día, en la opción estado dar clic en habilitar si selecciona deshabilitar quiere decir que se guarda la información pero no se la puede utilizar aun, una vez realizados todos los procesos se da clic en guardar (Ver Figura 30).

The screenshot shows the 'Sub - Grupos' window with the 'Guardar' button highlighted in blue. The window contains the following data entry fields:

- SubGrupo: PLOMERO
- Abreviatura: PLOM
- Descripción: OBRERO
- Estado: HABILITADO

A red arrow points from the text 'guardar' in the adjacent paragraph to the 'Guardar' button.

Figura 30

Si se desea modificar los datos seleccionamos la información dar clic en modificar (Ver Figura 31).

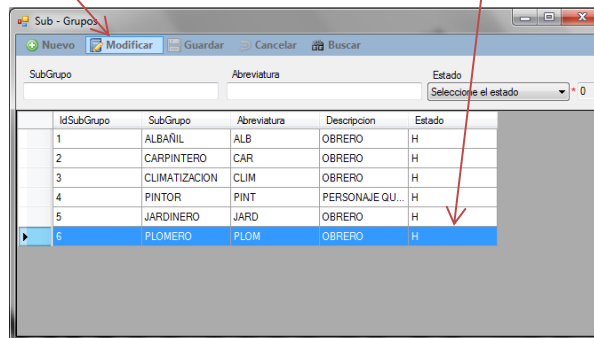


Figura 31

Aparece otra ventana en la cambiar o modificar, dar clic en la opción guardar, y listo (Ver Figura 32).

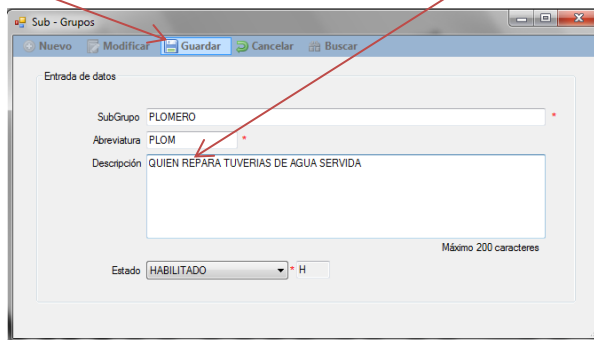


Figura 32

Con los nuevos cambios nos quedaría la información de esta forma. (Ver Figura 33).

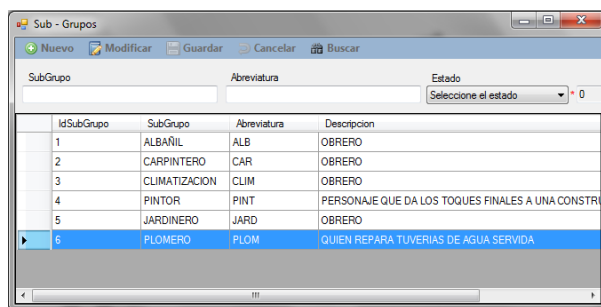


Figura 33

Para ingresar un producto de dar clic en la opción de mantenimiento al momento se despliega un menú en la que dar clic en Marca, (especifica la marca del producto) (Ver Figura 34).



Figura 34

Se visualiza la ventana en la que podemos ver todos los registros que se han realizado, para ingresar otra marca dar clic en nuevo (Ver Figura 35).

Id. Marca	Marca	Abreviatura	Descripción	Estado
1	SKIL	SKIL	HERRAMIENTA ELECTRICAS	H
3	BELLOTA	VELL	HERRAMIENTAS DE CONSTRUCCION	H
4	RUBI	RUB	HERRAMIENTAS DE CORTAR CERAMICA	H
5	STANLEY	STAN	HERRAMIENTA DE TODA CLASE EN CONSTRUCCION	H
6	BOSCH	BOSCH	HERRAMIENTA ELECTRICA	H
7	PLASTIGAMA	PLTA	MARCA DE TUBO PVC	H
8	BLACK&DECKER	BLACK&DECK	HERRAMIENTA ELECTRICA	H
9	SIKA	SIKA	MARCA DE PRODUCTOS QUE SE UTILIZA EN CONSTRUCCIONES	H
10	CEMENTO ROC...	CEM-ROC-50	MATERIAL DE CONSTRUCCION	H
11	DEWALT	DEW	ERRAMIENTAS ELECTRICAS UTILIZADAS CASI EN TODO TRABAJO	H

Figura 35

En este formulario ingresar la marcas del producto, se abrevia si se desea se le da una descripción o se puede realizarlo otro día, en la opción **Estado** dar clic se despliega un menú, seleccionar habilitar esta opción permite visualizar la información, en caso de que se opte por deshabilitado igualmente se guarda pero no se la puede utilizar aun, una vez realizados todos los procesos se da clic en guardar (Ver Figura 36).

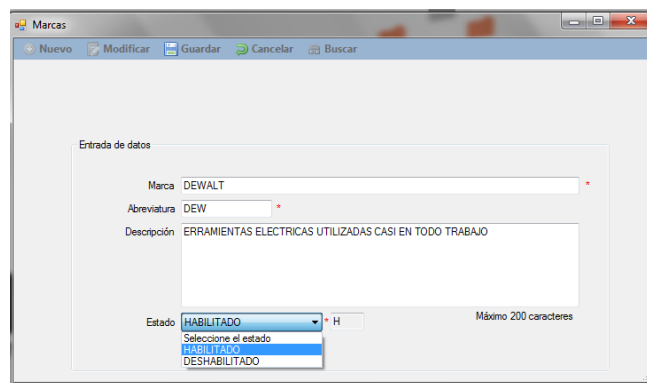


Figura 36

Para modificar los datos seleccionar la información y dar clic en modificar (Ver Figura 37).

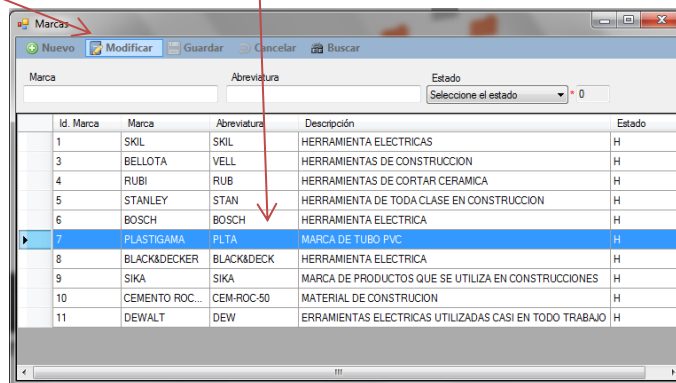


Figura 37

Aparece otra ventana para cambiar o modificar, dar clic en la opción guardar, y listo (Ver Figura 38).

Entrada de datos

Marca: PLASTIGAMA

Abreviatura: PLTA

Descripción: MARCA DE TUBO PVC QUE SE UTILIZA EN LAS AGUAS SERVIDAS

Estado: HABILITADO H Máximo 200 caracteres

Figura 38

Con los nuevos cambios nos quedaría de la siguiente manera (Ver Figura 39).

Id. Marca	Marca	Abreviatura	Descripción	Estado
1	SKIL	SKIL	HERRAMIENTA ELECTRICAS	H
3	BELLOTA	VELL	HERRAMIENTAS DE CONSTRUCCION	H
4	RUBI	RUB	HERRAMIENTAS DE CORTAR CERAMICA	H
5	STANLEY	STAN	HERRAMIENTA DE TODA CLASE EN CONSTRUCCION	H
6	BOSCH	BOSCH	HERRAMIENTA ELECTRICA	H
7	PLASTIGAMA	PLTA	MARCA DE TUBO PVC QUE SE UTILIZA EN LAS AGUAS SERVIDAS	H
8	BLACK&DECKER	BLACK&DECK	HERRAMIENTA ELECTRICA	H
9	SIKA	SIKA	MARCA DE PRODUCTOS QUE SE UTILIZA EN CONSTRUCCIONES	H
10	CEMENTO ROC...	CEM-ROC-50	MATERIAL DE CONSTRUCCION	H
11	DEWALT	DEW	ERRAMIENTAS ELECTRICAS UTILIZADAS CASI EN TODO TRABAJO	H

Figura 39

Línea.- Al seleccionar esta opción se despliega el siguiente formulario, el cual sirve para registrar los tipos de líneas que utilizan en material de bodega, el

formulario tiene operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 40).

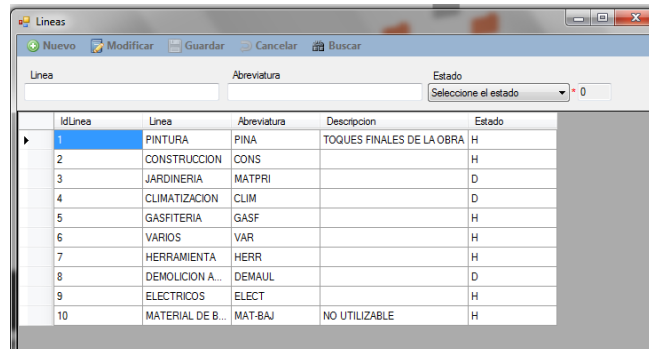


Figura 40

Éste formulario mediante el botón **Nuevo** permite ingresar nuevas líneas para su utilización en bodega y con el botón Guardar permite agregarlos al sistema.

Con el botón **Modificar** permite corregir errores que se hayan originado al momento de crear un nuevo tipo de línea con su descripción.

Con el botón **Buscar** permite visualizar todos los tipos de líneas que están registrados en el sistema.

Departamento.- Al seleccionar esta opción se despliega el siguiente formulario, el cual sirve para registrar los tipos de departamentos, done se realizan los trabajos y su vez también solicitan los mismos el formulario tiene las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 41).

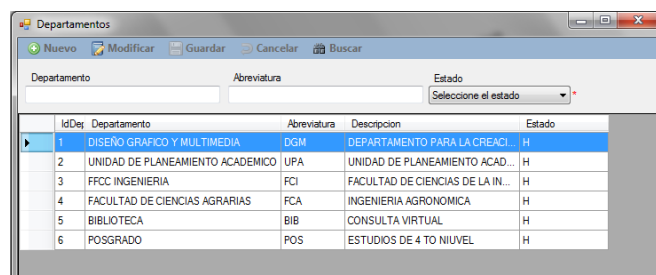


Figura 41

Éste formulario mediante el botón **Nuevo** permite registrar otro departamentos para su utilización en bodega y con el botón Guardar permite agregarlos al sistema.

Con el botón **Modificar** permite corregir errores que se hayan originado al momento de crear el nuevo registro.

Con el botón **Buscar** permite visualizar todos los departamentos que estén registrados en el sistema.

Personas / Entidades.- Al seleccionar esta opción se despliega el siguiente formulario de **Ingreso de Personal**, tanto de servicio que elaboran en la Unidad de Servicios Generales de la Universidad Técnica Estatal de Quevedo, como los departamentales quienes reciben los trabajos, el formulario tiene las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 42).

The screenshot shows a web application window titled "Ingreso de Personal". At the top, there are five buttons: "Nuevo", "Modificar", "Guardar", "Cancelar", and "Buscar". Below the buttons are input fields for "DNI", "Nombre", and "Estado" (with a dropdown menu). To the right, there is a section for "Ubicación Geográfica" with dropdown menus for "País", "Provincia", and "Ciudad". The main part of the window is a table with the following data:

Id	Tercero	DNI	Nombre	Dirección	Ciudad	Teléfono	Email	Ocupación	Estado
41		0800795445	VICTRO HUGO...	EL GUAYACAN	QUEVEDO	0997938261		ADMIN	H
42		1304055013	ANGEL MANUE...	LOS VERJELES	QUEVEDO	0992052223		JEFE DE BODEGA	H
43		1302062839	JOSE MANUEL...	LAS MARVINAS	QUEVEDO			TECNICO GASFI...	H
44		1204838567	DANNI DANIEL...	LOS PINOS	QUEVEDO			JARDINERO	H
45		1200955423	GALO TEOBALD...	EL PITAL	QUEVEDO			ALBAÑIL	H
46		1706718002	JULIO HUMBER...	LA VENUZ	MACHALA	0985264404		TECNICO GASFI...	H
47		1200636445	MAXIMO JAVIER...	EL PITAL	QUEVEDO			JARDINERO	H
48		1202972673	NEIL FRANCISC...	4 MANGAS	QUEVEDO			JARDINERO	H
49		0904527025	FRANKLIN RAF...	LAS MARVINAS	QUEVEDO			TECNICO ELEC...	H
50		1204762635	FELIX WILMER	GUAYACAN	QUEVEDO			TECNICO ELEC...	H
51		1203598535	HECTOR TOMA...	LOS CHAPULOS	QUEVEDO	0954034568	hectotom1000@...	JARDINERO	H
52		1308330875	STALIN CARRE...	LOS GUAYACAN...	QUEVEDO	0987656898		DIRECTOR	H
53		1203539034	GARY RAMIREZ	LAS PALMAS	QUEVEDO	0912344655		DECANO	H
54		1205719345	LEONARDO VA...	SAN CAMILO	QUEVEDO			JEFE DE TRANS...	H
55		1205781279	PABLO PALMA	LOS LAGOS	QUEVEDO			PROOVEDOR D...	H

Figura 42

Éste formulario mediante el botón **Nuevo** permite ingresar otra persona, tiene que ingresar el número de cedula de identidad sus nombres y apellidos dirección teléfono si tiene fax si tiene email si tiene una pequeña observación la ocupación si tiene otro contacto, en estado dar clic en habilitado para que el sistema lo mantenga siempre visible, también debe seleccionar las ubicaciones Geográficas como es el País de origen Provincia y Ciudad y si queremos agregarlos como proveedor seleccionamos el cuadrito, el que se activa como un visto y por ultimo damos clic en el botón Guardar permitiéndonos agregarlo al sistema. (Ver Figura 43).

The screenshot shows a web browser window titled "Ingreso de Personal". The interface includes a menu bar with buttons for "Nuevo", "Modificar", "Guardar", "Cancelar", and "Buscar". The main form area is titled "Entrada de datos" and is divided into several sections:

- Personal Data:** Fields for DNI (120398835), Nombre (HECTOR TOMAS REINA CHENCHE), Dirección (LOS CHAPULOS), Teléfonos (0954034568), Fax, and Email (hectotom1000@hotmail.com).
- Ubicación Geográfica:** Dropdown menus for País (ECUADOR), Provincia (LOS RIOS), and Ciudad (GUEVEDO).
- Observación:** A text area containing "ENTRADA 8H00, SALIDA ALZ 12H00, ENTRADA ALZ 12H30, SALIDA 16H30".
- Es proveedor?:** A checkbox that is currently unchecked.
- Occupation and Contact:** Fields for Ocupación (JARDINERO) and Contacto.
- Estado:** A dropdown menu set to "HABILITADO".

Figura 43

Con el botón **Modificar** permite corregir errores que se hayan originado al momento de crear un nuevo registro.

Con el botón **Buscar** permite visualizar el número de cédula registrado en el sistema.

Cargo.- Al seleccionar esta opción se despliega el siguiente formulario, el cual sirve para registrar los tipos de cargo asignados a cada usuario, este formulario tiene las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 44).



The screenshot shows a window titled "Cargos" with a toolbar containing buttons for "Nuevo", "Modificar", "Guardar", "Cancelar", and "Buscar". Below the toolbar are three input fields: "Cargo", "Abreviatura", and "Estado" (with a dropdown menu labeled "Seleccione el estado"). Below the form is a table with the following data:

	IdCarg	Cargo	Abreviatura	Descripcion	Estado
▶	1	BODEGUERO	BOD	PERSONA ENCARGADA DE LA ...	H
	2	JEFE DE BODEGA -USG	JEFBOD	JEFE DE LA BODEGA DE LA UNI...	H
	3	DIRECTOR DEPARTAMENTAL	DIR-DEPAR	DIRECTOR DE FACULTADES	H
	4	DECANO	DECA	AUTORIDADES DEPARTAMENT...	H
	5	SUB DECANO	SUB DECA	AUTORIDADES DEPARTAMENT...	H
	6	COORDINADOR U.S.G.	COORD	CORDINA LAS ACTIVIDADES E...	H

Figura 44

Éste formulario mediante el botón **Nuevo** permite ingresar nuevas cargas para ser asignadas al personal y con el botón Guardar permite agregarlos al sistema.

Con el botón **Modificar** permite corregir errores que se hayan originado al momento de crear una nuevo cargas.

Con el botón **Buscar** permite visualizar todas las cargas que estén registradas en el sistema.

Asignación de Cargo.- Al seleccionar esta opción se despliega el siguiente formulario, el cual se muestra en blanco es decir sin registros (Ver Figura 45), para lo cual se debe dar clic en el signo más (+), mostrando el formulario de Ingreso de Personal (Ver Figura 46), damos doble clic sobre la persona que queremos asignarle el cargo y automáticamente se muestra sobre la línea donde dice Seleccione una persona (Ver Figura 47).

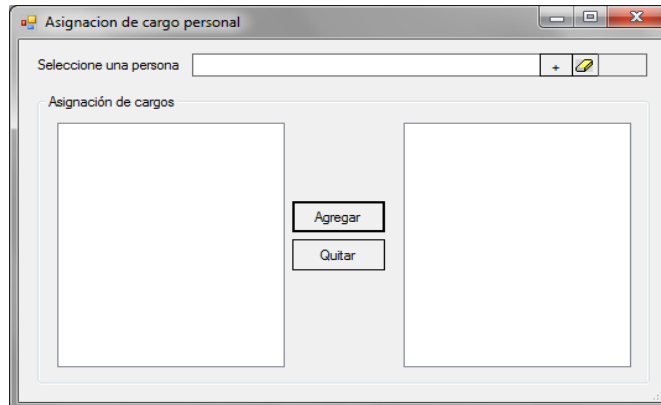


Figura 45

	Id. Tercero	DNI	Nombre	Dirección	Ciudad	Teléfono	Email	Ocupación	Estado
▶	41	0800795445	VICTRO HUGO ...	EL GUAYACAN	QUEVEDO	0997938261		ADMIN	H
	42	1304055013	ANGEL MANUE ...	LOS VERJELES	QUEVEDO	0992052223		JEFE DE BODEGA	H
	43	1302062839	JOSE MANUEL ...	LAS MARVINAS	QUEVEDO			TECNICO GASFI...	H
	44	1204838567	DANNI DANIEL ...	LOS PINOS	QUEVEDO			JARDINERO	H
	45	1200955423	GALO TEOBALD ...	EL PITAL	QUEVEDO			ALBAÑIL	H
	46	1706718002	JULIO HUMBER...	LA VENUZ	MACHALA	0985264404		TECNICO GASFI...	H
	47	1200636445	MAXIMO JAVIER...	EL PITAL	QUEVEDO			JARDINERO	H
	48	1202972673	NEIL FRANCISC...	4 MANGAS	QUEVEDO			JARDINERO	H
	49	0904527025	FRANKLIN RAF...	LAS MARVINAS	QUEVEDO			TECNICO ELEC...	H
	50	1204762635	FELIX WILMER	GUAYACAN	QUEVEDO			TECNICO ELEC...	H
	51	1203598535	HECTOR TOMA...	LOS CHAPULOS	QUEVEDO	0954034568	hectotom1000@...	JARDINERO	H
	52	1308330875	STALIN CARRE...	LOS GUAYACAN...	QUEVEDO	0987656898		DIRECTOR	H
	53	1203539034	GARY RAMIREZ	LAS PALMAS	QUEVEDO	0912344655		DECANO	H
	54	1205719345	LEONARDO VA...	SAN CAMILO	QUEVEDO			JEFE DE TRANS...	H
	55	1205781279	PABLO PALMA	LOS LAGOS	QUEVEDO			PROOVEDOR D...	H

Figura 46

En la parte derecha se encuentran todas las asignaciones, se debe seleccionar una u otra ya que una misma persona puede tener algunos cargos y damos clic en agregar y serramos el formulario.

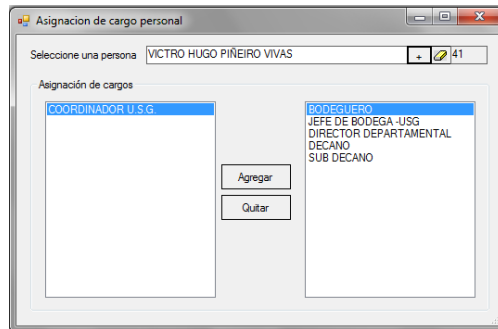


Figura 47

- **Menú Entradas**

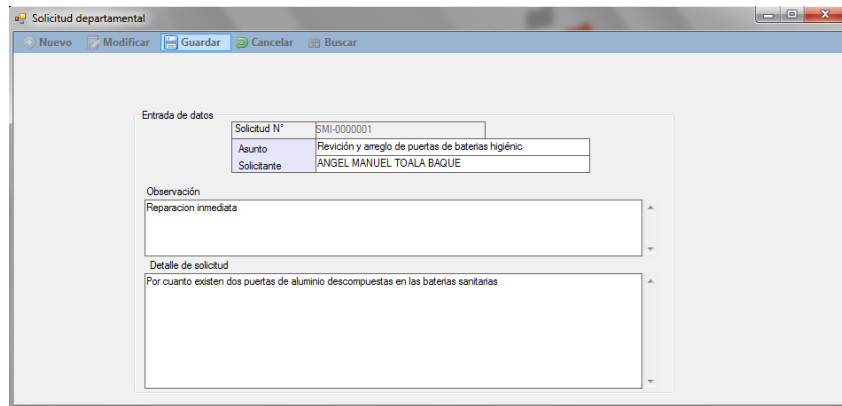
Al seleccionar el menú Entrada se despliega diferentes opciones como son:

- 1.- Solicitud departamental
- 2.- Solicitud de contratación
- 3.- Compras
- 4.- Recibir compras
- 5.- Registro de labores del personal (Ver Figura 48).



Figura 48

Solicitud Departamental.- Al seleccionar esta opción se despliega el siguiente formulario, el cual sirve para realizar los tipos de pedidos de obras que solicite el jefe departamental de la Universidad Técnica Estatal de Quevedo, a la Unidad de Servicio Generales de la misma institución, el formulario realiza las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 49).



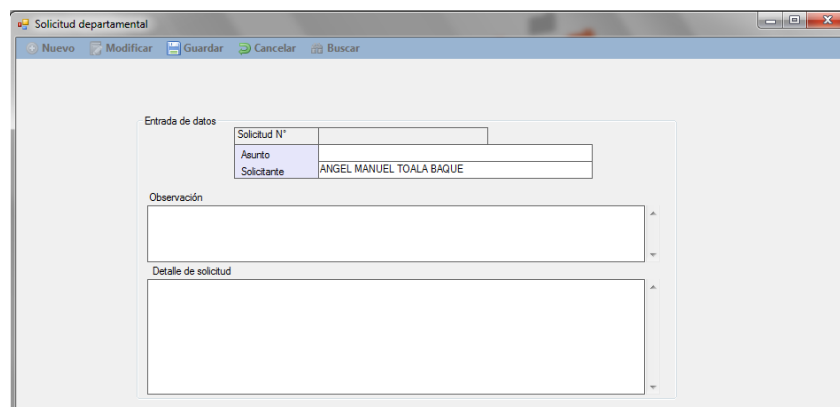
The screenshot shows a web application window titled "Solicitud departamental". At the top, there is a navigation bar with buttons for "Nuevo", "Modificar", "Guardar", "Cancelar", and "Buscar". The main content area is divided into three sections:

- Entrada de datos:** A table with three rows:

Solicitud N°	SM-000001
Asunto	Revisión y arreglo de puertas de baterías higiénic
Solicitante	ANGEL MANUEL TOALA BAQUE
- Observación:** A text area containing the text "Reparacion inmediata".
- Detalle de solicitud:** A text area containing the text "Por cuanto existen dos puertas de aluminio descompuestas en las baterías sanitarias".

Figura 49

Mediante el botón **Nuevo** permite ingresar nuevas solicitudes, el jefe departamental por lo consiguiente aparece automáticamente como solicitante, se describe las observaciones el detalle y se guarda al sistema. (Ver Figura 50).



This screenshot shows the same "Solicitud departamental" form as in Figure 49, but with the input fields for "Observación" and "Detalle de solicitud" empty. The "Entrada de datos" section remains filled with the same information: Solicitud N° SM-000001, Asunto Revisión y arreglo de puertas de baterías higiénic, and Solicitante ANGEL MANUEL TOALA BAQUE.

Figura 50

De igual manera si se desea **Modificar** se lo realiza corrigiendo algún error que se hayan originado al momento de crear la solicitud.

Con el botón **Buscar** permite visualizar las solicitudes mediante la Solicitud N° (ejemplo) SMI-0000002 que se allá registrado en el sistema. (Ver Figura 52).

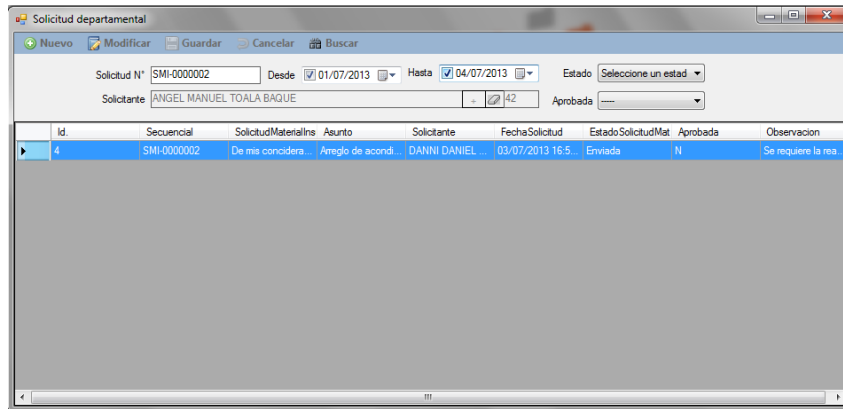


Figura 51

Solicitud de Contratación.- Al seleccionar esta opción se despliega el siguiente formulario, el cual sirve para realizar los tipos de pedidos de material mediante solicitudes recibidas de los jefes departamentales de la Universidad Técnica Estatal de Quevedo, para realizar los respectivos arreglos, el formulario realiza las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 52).

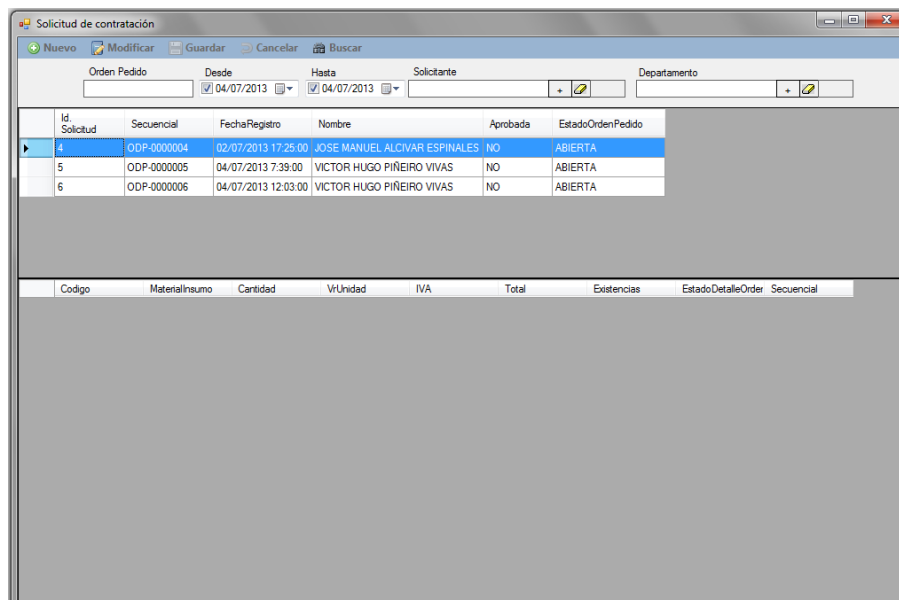


Figura 52

Mediante el botón **Nuevo** permite ingresar nuevas solicitudes de contratación, esto solo lo puede realizarlo el ingeniero Víctor Piñero Jefe de la Unidad de Servicios Generales. (Ver Figura 53).


Código	Material / Insumo	Cantidad	Vr. Unidad	Subtotal	Existencias
M.T.F. 0001	TUBOS FLUORE...	25	12,00	300,00	0,00
E-T-003	TUBOS FLUORE...	15	0,00	0,00	0,00

Figura 53

A continuación se describe los detalles de cómo realizarlo.

- 1.- En **Solicitud** damos clic para ingresa el solicitante mencionado.
- 2.- En **Asunto** se detalla el motivo de reparación.
- 3.- En observaciones la necesidad de lo solicitado.
- 4.- En **La adquisición consta en el PAC del año en curso** se escoge una.
- 5.- En **Parámetros Contratación** damos clic en la que aparece un formulario de requerimiento, en la cual escogemos las opciones requeridas. (Ver Figura 54).

Figura 54

6.- En **Material / Insumo** damos clic sobre este icono  en que aparece un formulario de **Inventario**. (Ver Figura 55).

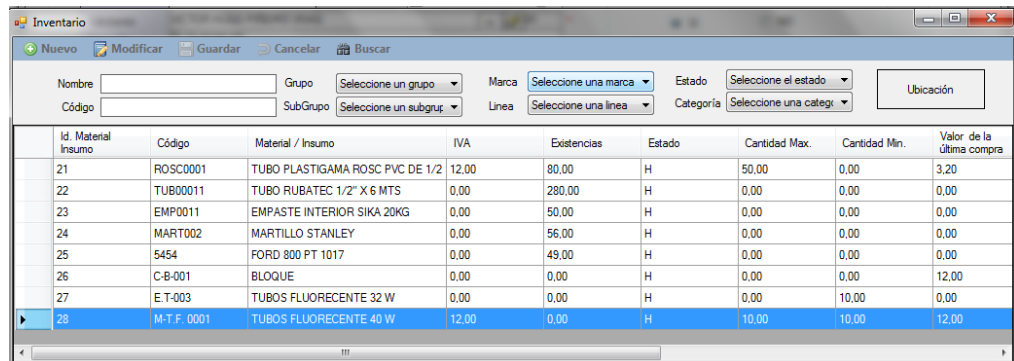
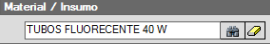



Figura 55

Damos doble clic sobre el material que se va a solicitar, de manera que se ingresa en la casilla de **Material Insumo**  en la siguiente casilla se ingresa la cantidad del pedido y se da clic en el (+)  para se ingrese la cantidad del pedido.

Con el botón **Modificar** nos permite corregir errores que se hayan realizados al momento de ingresar información, los pasos no cambian, por lo que se puede realizar con los pasos anteriores de ingreso y damos clic en guardar.

Compras.- Al seleccionar esta opción se despliega el siguiente formulario de **Orden de Compras**, el cual sirve para realizar los pedidos de material que se requieren para cumplir con las necesidades de reparaciones de daños ocasionados por desperfectos o por estar en mal estados es la Universidad Técnica Estatal de Quevedo, el formulario realiza las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 56).

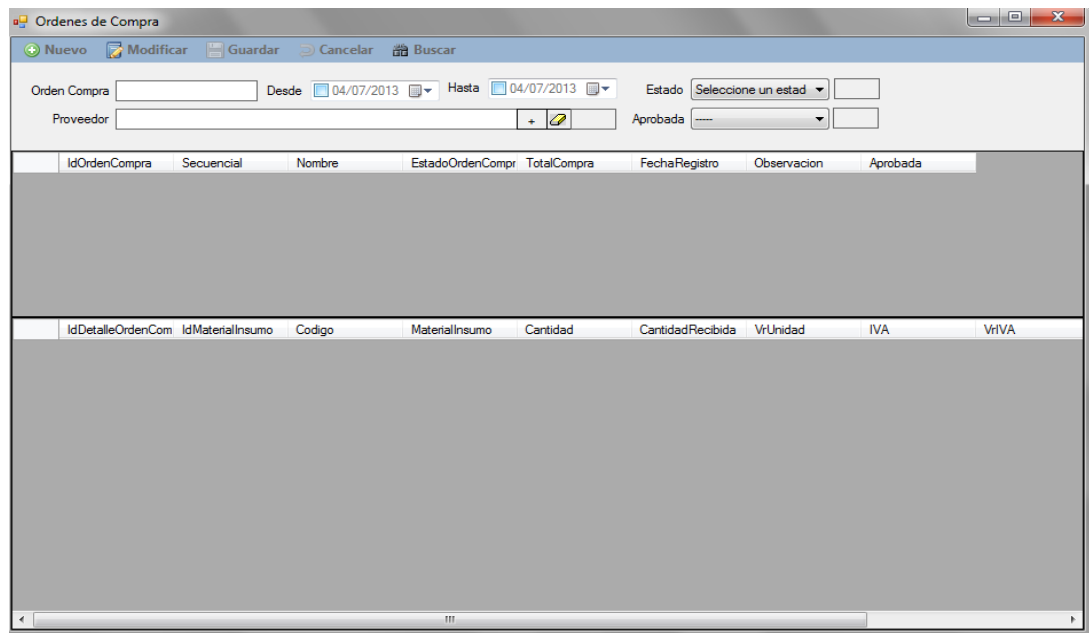



Figura 56

Mediante el botón **Nuevo** permite ingresar nuevas Órdenes de Compras, esto solo lo puede de la siguiente manera.

1.- En proveedor damos un clic en el (+), en la cal muestra un listado de usuarios en la que damos doble clic sobre el proveedor, el cual se ingresa al listado, el resto se llena acorde los requerimientos del pedido, en **Material / Insumos** y en **Cantidad**, damos clic sobre este icono  en que aparece el formulario de Inventario en el que se encuentra el listado de material, de la misma forma damos doble clic para que se agregue a la lista de pedidos, le especificamos la cantidad y damos clic en el (+) para que se agregue al listado de la derecha. (Ver Figura 57).

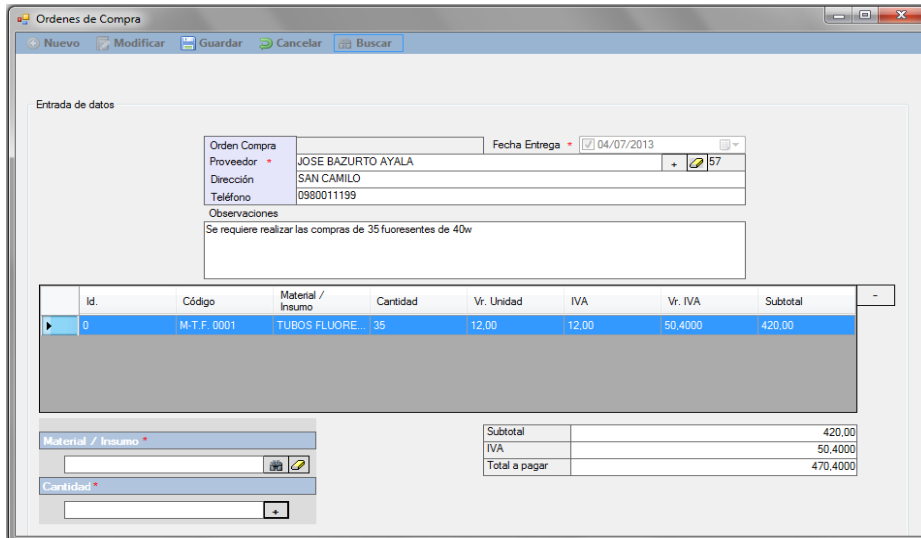


Figura 57

Con el botón **Modificar** nos permite corregir errores que se hayan realizados al momento de ingresar información, los pasos no cambian, por lo que se puede realizar con los pasos anteriores de ingreso y damos clic en guardar. (Ver Figura 58).

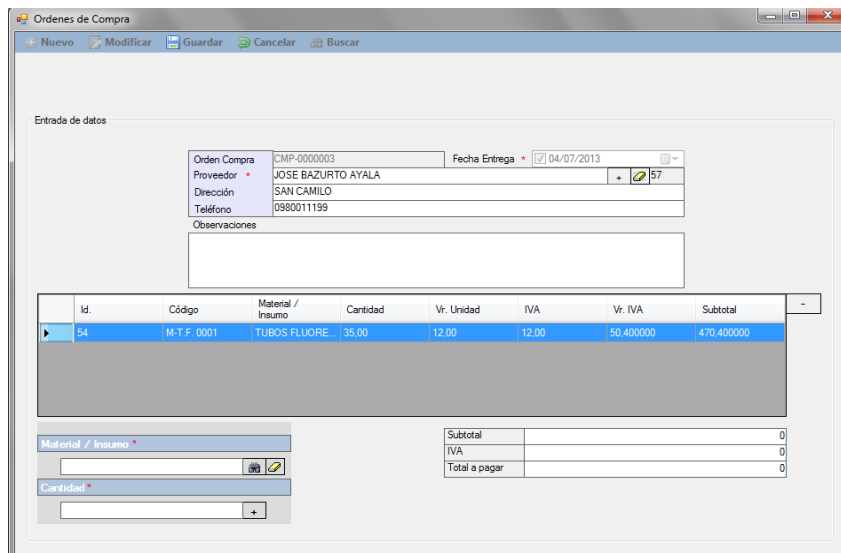


Figura 58

Recibir Compras.- Al seleccionar esta opción se despliega el siguiente formulario de **Recibir Materiales / Insumo a Bodega**, el cual sirve para realizar el ingreso de material a bodega que se requieren para cumplir con las necesidades de reparaciones de daños ocasionados por desperfectos o por estar en mal estados es la Universidad Técnica Estatal de Quevedo, (Ver Figura 59).

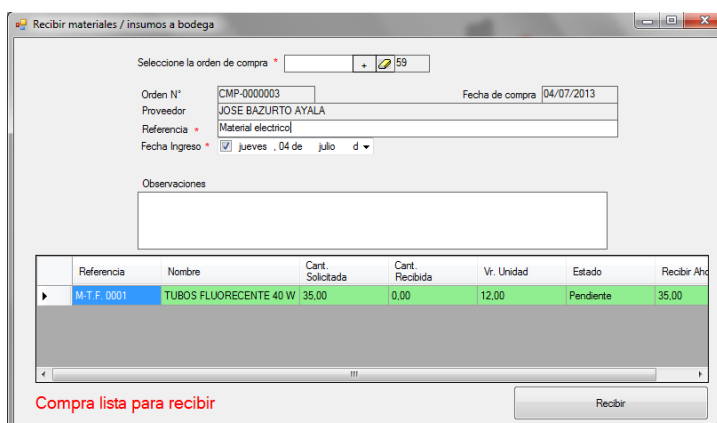


Figura 59

Los pasos para recibir el material son:

1.- En la opción **Seleccione la orden de compra** damos clic en el (+), en el que aparece el formulario orden de compras, damos doble clic sobre la compra para que se ingresen la información, se escribe la referencia, se activa la fecha, en observaciones se puede escribir algún detalle del ingreso, damos clic en recibir y luego en terminar, quiere decir que los ingresos se realizaron bien. (Ver Figura 60).

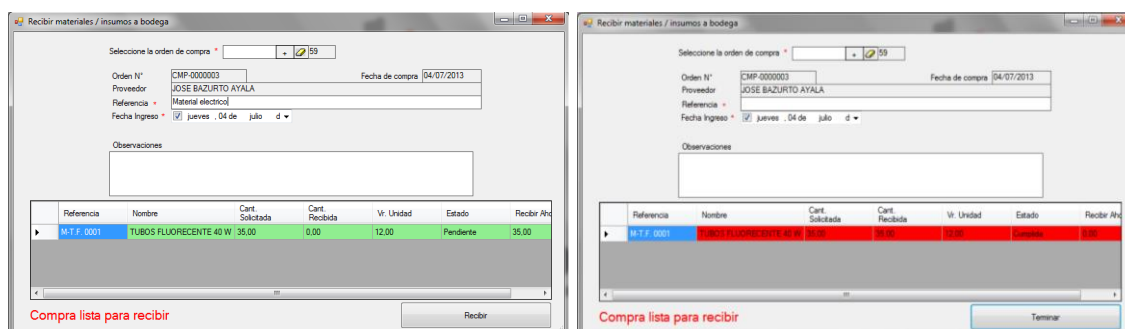


Figura 60

Registro de labores del personal.- Al seleccionar esta opción se despliega el siguiente formulario de **Registro de labores del personal**, el cual sirve para realizar el ingreso de las actividades realizadas del personal que laboran semanales de la Unidad de Servicios Generales. (Ver Figura 61).

Id. Detalle Actividad	Id. Colaborador	Colaborador	Fecha
5	46	JULIO HUMBERTO CARABAJA PERALTA	02/07/201
0	43	JOSE MANUEL ALCIVAR ESPINALES	02/07/201
0	45	GALO TEOBALDO CANO HERRERA	28/05/201
0	48	NEIL FRANCISCO CERCADO COBEÑA	28/05/201

Figura 61

- **Menú Salidas**

Al seleccionar el menú **Salidas** se despliega diferentes opciones como son:

1.- Consignaciones

2.- Consumo

(Ver Figura 62).



Figura 62

Consignaciones.- Al seleccionar esta opción se despliega el siguiente formulario, el cual sirve para realizar los tipos de pedidos de material mediante solicitudes recibidas de los jefes departamentales de la Universidad Técnica Estatal de Quevedo, para realizar los respectivos arreglos, el formulario realiza las operaciones como Nuevo, Modificar, Guardar, Cancelar y Buscar. (Ver Figura 63).

Id.	Código	Material / Insumo	Cantidad	Vr. Unidad	IVA	Vr. IVA	Subtotal
0	ROSC0001	TUBO PLASTIG...	220	3,20	12,00	84.4800	704,00

Figura 63

Consumo.- Al seleccionar esta opción se despliega el formulario, en el cual sirve para realizar los consumos que se llevan semanales, para poder llevar el control de material que ingresan a bodega. (Ver Figura 64).

The screenshot shows a software window titled 'Consumo'. It contains several data entry fields and a table. The fields include 'Consumo N°', 'Quien Recibe' (GARY RAMIREZ), 'Departamento', 'Referencia de Solicitud' (Areglo de acondicionador de aire), and 'Fecha Consumo' (viernes, 05 de julio de 2013). There are also dropdown menus for 'Material / Insumo' (TUBOS FLUORECENTE 40 W) and 'Cantidad' (35). A table with columns 'Id.', 'Código', 'Material / Insumo', 'Cantidad', 'Vr. Unidad', 'IVA', 'Vr. IVA', and 'Subtotal' is displayed. The table contains one row with the following data: Id. 0, Código C-T-R 0005, Material / Insumo TUBOS RIVAL, Cantidad 23, Vr. Unidad 8.00, IVA 0.00, Vr. IVA 0.0000, Subtotal 184.00. Below the table are 'Guardar' and 'Cancelar' buttons, and a summary table with values 184.00, 0.0000, and 184.0000.

Id.	Código	Material / Insumo	Cantidad	Vr. Unidad	IVA	Vr. IVA	Subtotal
0	C-T-R 0005	TUBOS RIVAL	23	8.00	0.00	0.0000	184.00

Figura 64

- **Menú Almacén**

Al seleccionar el menú Almacén se despliega diferentes opciones como son:

- 1.- kardex
 - 2.- Inventario
- (Ver Figura 65).



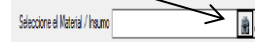
Figura 65

Al seleccionar la opción Kardex se presenta un formulario que permite registrar en el sistema los datos de cada ingreso / egreso que se realiza durante las labores semanales. (Ver Figura 66).

The 'Kardex' window contains a search bar at the top with the text 'Seleccione el Material / Insumo'. Below it is a table with columns: 'Codigo', 'Stock Mínimo', 'Existencias', 'Vr. Compra', 'Grupo', 'SubGrupo', 'Marca', and 'Fecha Ult. Cons.'. At the bottom, there is a table with columns: 'IdKardex', 'FechaRegistro', 'Concepto', 'Entradas', 'Salidas', 'Saldo', 'Precio', and 'Nota'. Two buttons, 'Recibir' and 'Retirar', are located at the bottom of the window.

Figura 66

La forma de ingresar datos al kardex es de forma sencilla ya que una vez que se obtengan los productos o materiales ingresados al sistema se ahorrara tiempo, para ingresar los datos damos clic sobre este icono, al momento aparece otro formulario con la información (Ver Figura 67).



The 'Inventario' window displays a list of materials with the following columns: 'Id. Material Insumo', 'Código', 'Material / Insumo', 'IVA', 'Existencias', 'Estado', 'Cantidad Max.', 'Cantidad Min.', 'Valor de la última compra', 'Grupo', 'SubGrupo', 'Linea', 'Marca', and 'Catego'. The table contains 8 rows of data.

Id. Material Insumo	Código	Material / Insumo	IVA	Existencias	Estado	Cantidad Max.	Cantidad Min.	Valor de la última compra	Grupo	SubGrupo	Linea	Marca	Catego
21	ROSC0001	TUBO PLASTIG...	12.00	80.00	H	50.00	0.00	3.20	INSUMOS	CARPINTERO	GASFITERIA	PLASTIGAMA	NORMA
22	TUB00011	TUBO RUBATE...	0.00	280.00	H	0.00	0.00	0.00	INSUMOS	CARPINTERO	GASFITERIA	BLACK&DECKER	NORMA
23	EMP0011	EMPASTE INTE...	0.00	50.00	H	0.00	0.00	0.00	INSUMOS	CARPINTERO	PINTURA	SIKA	NORMA
24	MART002	MARTILLO STA...	0.00	56.00	H	0.00	0.00	0.00	MATERIALES DE REPUESTOS Y ACCESORIOS	CARPINTERO	HERRAMIENTA	STANLEY	NORMA
25	C-T-R 0005	TUBOS RIVAL	0.00	49.00	H	0.00	0.00	8.00	MATERIAL DE CONSTRUCCION	PLOMERO	CONSTRUCCION	PLASTIGAMA	NORMA
26	C-B-001	BLOQUE	0.00	0.00	H	0.00	0.00	12.00	MATERIALES DE REPUESTOS Y ACCESORIOS	CARPINTERO	CONSTRUCCION	PLASTIGAMA	NORMA
27	E.T-003	TUBOS FLUORE...	0.00	0.00	H	0.00	10.00	0.00	MATERIAL DE CLIMATIZACION	ELECTRICO	ELECTRICOS	SKIL	NORMA
28	M-T.F. 0001	TUBOS FLUORE...	12.00	35.00	H	10.00	10.00	12.00	MATERIAL DE ELECTRICO	ELECTRICO	ELECTRICOS	DAYLIGHT	NORMA

Figura 67

Al dar doble clic sobre el registro seleccionado, se llenan todos los campos del formulario kardex. (Ver Figura 68).

The 'Kardex' window now shows the form populated with data from the selected record (row 22 in Figure 67). The search bar contains 'TUBO RUBATEC 1/2" X 6 MTS'. The table below has one row with the following data: '27', '06/06/2013 17:5...', 'Inventario Inc...', '300.00', '0.00', '0.00', '0.00', and 'asaca'.

Figura 68

Luego damos clic en recibir mostrándonos un mensaje que dice El Kardex ha sido actualizado correctamente y damos clic en aceptar y cerramos. (Ver Figura 69).

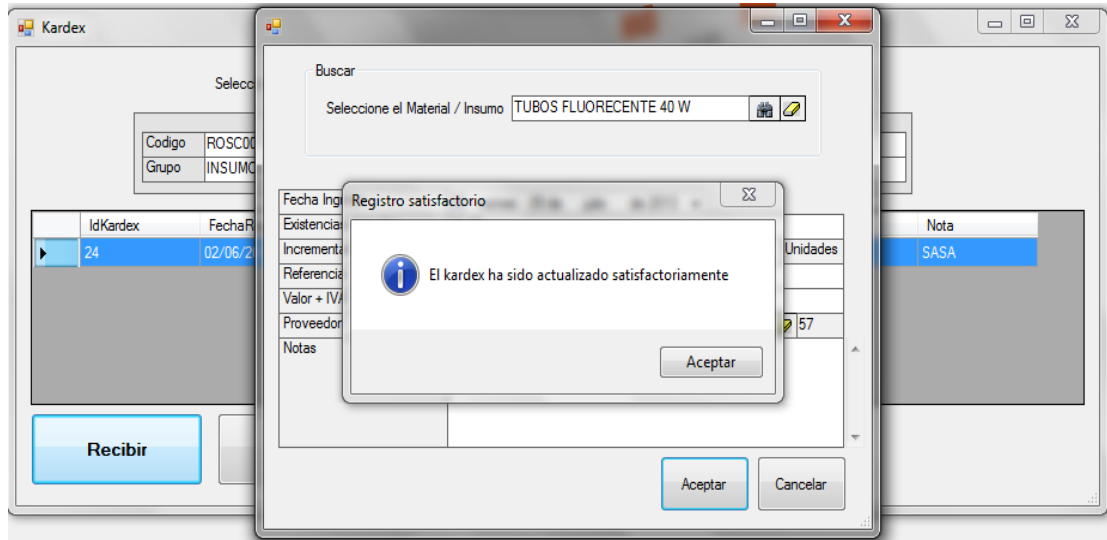


Figura 69

Inventario.- Al seleccionar esta opción se despliega el formulario, en el cual visualizamos el material existente en bodega. (Ver Figura 70).

Id. Material Insumo	Código	Material / Insumo	IVA	Existencias	Estado	Cantidad Max.	Cantidad Min.	Valor de la última compra	Grupo	SubGrupo	Línea	Marca	Categoría
21	ROSC001	TUBO PLASTIG...	12.00	80.00	H	90.00	0.00	3.20	INSUMOS	CARPINTERO	GASFITERIA	PLASTIGAMA	NORMAL
22	TUB00011	TUBO RUBATE...	0.00	280.00	H	0.00	0.00	0.00	INSUMOS	CARPINTERO	GASFITERIA	BLACK&DECKER	NORMAL
23	EMP0011	EMPASTE INTE...	0.00	50.00	H	0.00	0.00	0.00	INSUMOS	CARPINTERO	PINTURA	SIKA	NORMAL
24	MART002	MARTILLO STA...	0.00	56.00	H	0.00	0.00	0.00	MATERIALES D...	CARPINTERO	HERRAMIENTA	STANLEY	NORMAL
25	C-T-R 0005	TUBOS RIVAL	0.00	49.00	H	0.00	0.00	8.00	MATERIAL DE C...	PLOMERO	CONSTRUCCION	PLASTIGAMA	NORMAL
26	C-B-001	BLOQUE	0.00	0.00	H	0.00	0.00	12.00	MATERIALES D...	CARPINTERO	CONSTRUCCION	PLASTIGAMA	NORMAL
27	E-T-003	TUBOS FLUORE...	0.00	0.00	H	0.00	10.00	0.00	MATERIAL DE C...	ELECTRICO	ELECTRICOS	SKIL	NORMAL
28	M-T-F. 0001	TUBOS FLUORE...	12.00	35.00	H	10.00	10.00	12.00	MATERIAL DE E...	ELECTRICO	ELECTRICOS	DAYLIGHT	NORMAL
29	C-T-F. 40W	TUBOS FLUORE...	0.00	0.00	H	10.00	0.00	0.00	MATERIAL DE E...	ELECTRICO	ELECTRICOS	DAYLIGHT	NORMAL

Figura 70

- **Menú Informes**

Al seleccionar el menú **Informes** se despliega diferentes opciones como son:

- 1.- Órdenes de Pedido
- 2.- Órdenes de Compra
- 3.- Inventario

4.- Kardex (Ver Figura 70).

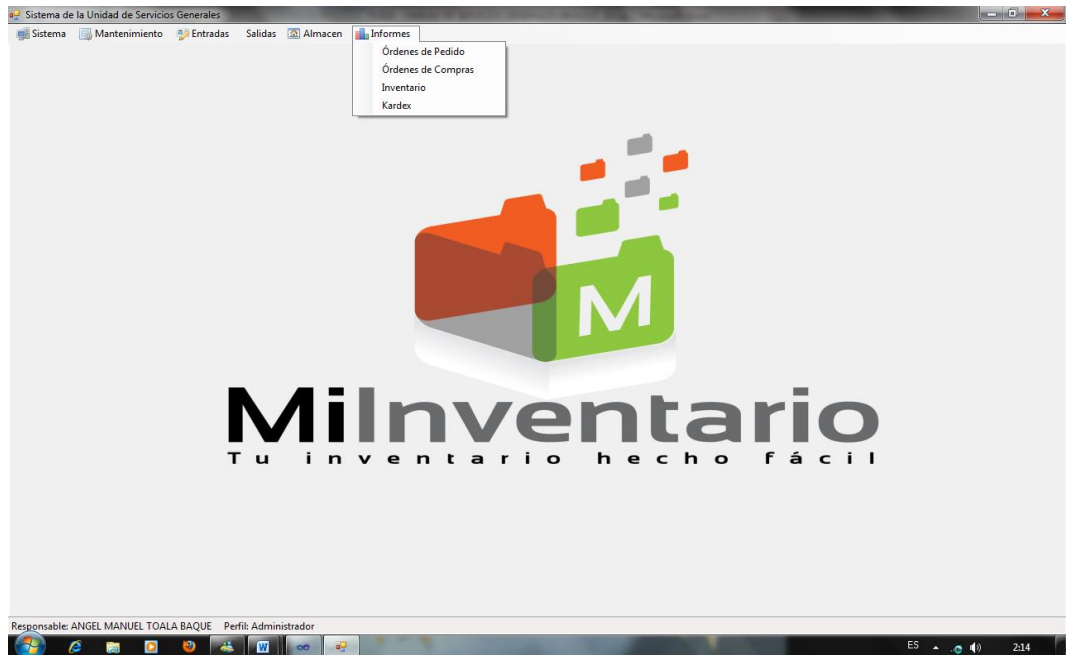


Figura 70

Al seleccionar la opción **Órdenes de Pedido** se presenta un formulario en el cual se tiene que ingresar la información, (Ver Figura 71), dando un clic sobre el (+), se abre otro formulario con la información requerida para ser ingresada, damos doble clic sobre la información para que se adhiera al formulario de órdenes de pedido (Ver Figura 72).

Figura 71

Solicitud de contratación

Nuevo Modificar Guardar Cancelar Buscar

Orden Pedido Desde 05/07/2013 Hasta 05/07/2013 Solicitante Departamento

Id. Solicitud	Secuencial	FechaRegistro	Nombre	Aprobada	EstadoOrdenPedi:
4	ODP-0000004	02/07/2013 17:2...	JOSE MANUEL ...	NO	ABIERTA
5	ODP-0000005	04/07/2013 7:39...	VICTOR HUGO ...	NO	ABIERTA
6	ODP-0000006	04/07/2013 12:0...	VICTOR HUGO ...	NO	ABIERTA

Codigo	MaterialInsumo	Cantidad	VrUnidad	IVA	Total	Existencias	EstadoDetalleOrder	Secuencial
M-T.F. 0001	TUBOS FLUORE...	25.00	12.00		300.00	0.00		
E.T-003	TUBOS FLUORE...	15.00	0.00		0.00	0.00		

Figura 72

Al momento que ingresa la Orden de Pedido se opaca el formulario y se da clic en General Reporte (Ver Figura 73).

Ordenes de Pedido

Buscar por orden de pedido

Orden de Pedido ODP-0000006 + 6

Filtros

Estado Seleccione un estado de orden de pedido

Fecha Inicio viernes . 05 de julio de 2013

Fecha Fin viernes . 05 de julio de 2013

Solicitante Departamento

Aprobada

SI

NO

Presentación

Gráfico

Detalle

Resumen

Generar Reporte

Figura 73

Este es el reporte de Orden de Pedido generado (Ver Figura 74).

Orden de Pedido

1 de 1

100%

Buscar | Siguiente

Universidad Técnica Estatal de Quevedo

Unidad de Servicios Generales (USG)
 1201143714
 Campus Manuel Haz Alvarez, KM 1 1/2 Vía a Santo Domingo de los Tsáchilas
 05287878
 Quevedo - Los Ríos - Ecuador
 compraspublicas@uteq.edu.ec | www.uteq.edu.ec

Orden de Pedido
ODP-000006

Quien Solicita VICTOR HUGO PIÑEIRO VIVAS

Aprobada SI **Fecha** 04/07/2013 12:03:00

Estado CERRADA **Departamento** BIBLIOTECA

Codigo	Material Insumo	Cantidad	Estado Detalle Orden Pedido
M-T.F. 0001	TUBOS FLUORECENTE 40 W	25,00	Pendiente
E.T-003	TUBOS FLUORECENTE 32 W	15,00	Pendiente

Página: 1

Figura 74

Al seleccionar la opción **Órdenes de Compra** se presenta un formulario en el cual se tiene que ingresar la información, (Ver Figura 75), dando un clic sobre el (+), se abre otro formulario con la información requerida para ser ingresada, damos doble clic sobre la información para que se adhiera al formulario de órdenes de compra (Ver Figura 76).

Órdenes de Compra

Buscar por orden de pedido

Orden de Compra +

Filtros

Estado

Fecha Inicio viernes, 05 de julio de 2013

Fecha Fin viernes, 05 de julio de 2013

Proveedor +

Aprobada SI NO

Presentación Gráfico Detalle Resumen

Generar Reporte

Figura 75

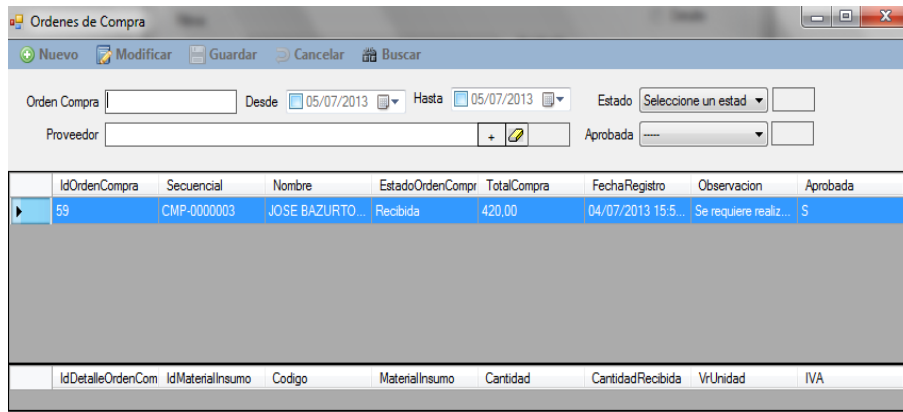


Figura 76

Al momento que ingresa la Orden de Compra se opaca el formulario y se da clic en General Reporte (Ver Figura 77).

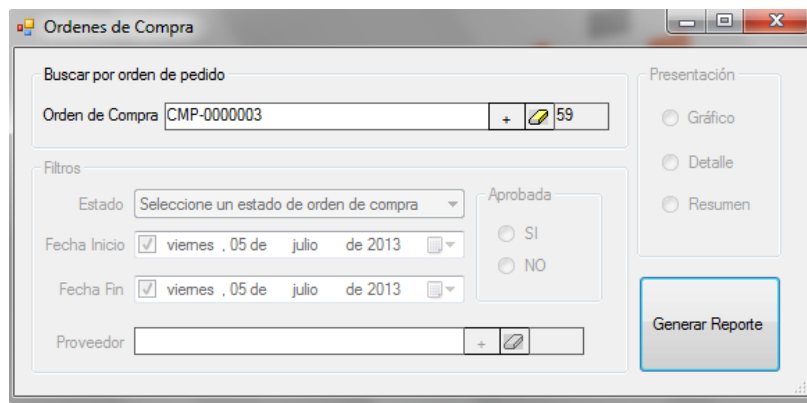


Figura 77

Este es el reporte de Orden de Compras generado (Ver Figura 78).

Ordenes de Compra

1 de 1

100%

Buscar | Siguiete

Universidad Técnica Estatal de Quevedo

Unidad de Servicios Generales (USG)
1201143714
Campus Manuel Haz Alvarez, KM 1 1/2 vía a Santo Domingo de los Tsachilas
05287878
Quevedo - Los Rios - Ecuador
compraspublicas@uteq.edu.ec | www.uteq.edu.ec

Orden de Compra
CMP-0000003

Proveedor JOSE BAZURTO AYALA

Aprobada S **Estado** Recibida

Codigo	Material Insumo	Cantidad	Estado Detalle Orden Compra
M-T.F. 0001	TUBOS FLUORECENTE 40 W	35,00	Cumplida

Página: 1

Figura 78

Al seleccionar la opción **Inventario** se presenta un formulario en el cual se tiene que escribir un nombre específico, seleccionar cada uno de los casilleros, también, si desea darle alguna presentación selecciona una de ellas y se da clic en General Reporte (Ver Figura 79).

Figura 79

Este es el reporte de Orden de Compras generado (Ver Figura 80).

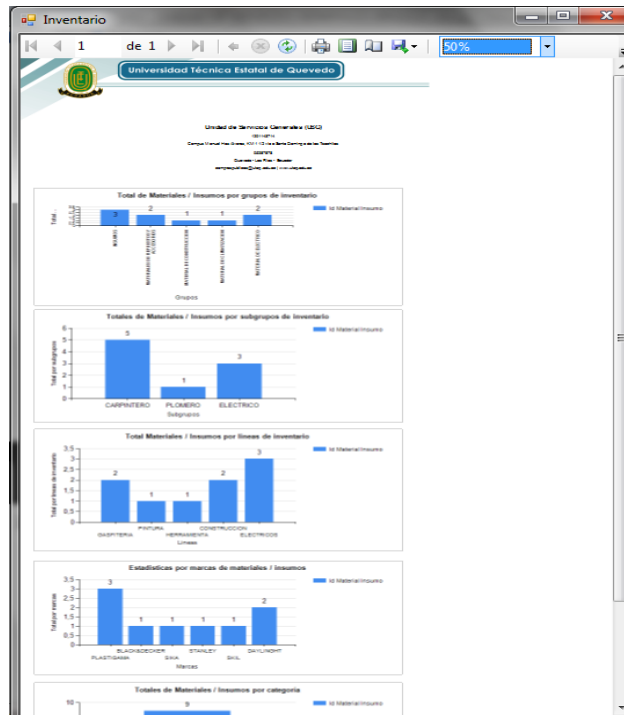


Figura 80

Al seleccionar la opción **Kardex** se presenta un formulario en donde tiene que, un clic sobre en este icono, se (Ver Figura 81), se abre otro formulario con la información requerida para ser ingresada, damos doble clic sobre la información para que se adhiera al formulario de Kardex (Ver Figura 82).

The 'Kardex' window contains a search bar at the top with the text 'Seleccione el Material / Insumo'. Below it is a table with the following fields:

Código	Stock Mínimo	Existencias	Vr. Compra
Grupo	SubGrupo	Marca	Fecha Ult. Cons.

There is also a 'Generar Reporte' button on the right side of the window.

Figura 81

The 'Inventario' window displays a list of inventory items. The table below shows the data for the first few items:

Id. Material Insumo	Código	Material / Insumo	IVA	Existencias	Estado	Cantidad Max.	Cantidad Min.	Valor de la última compra	Grupo
21	ROSC0001	TUBO PLASTIG...	12,00	80,00	H	50,00	0,00	3,20	INSUMOS
22	TUB00011	TUBO RUBATE...	0,00	280,00	H	0,00	0,00	0,00	INSUMOS
23	EMP0011	EMPASTE INTE...	0,00	50,00	H	0,00	0,00	0,00	INSUMOS
24	MART002	MARTILLO STA...	0,00	56,00	H	0,00	0,00	0,00	MATERIALES D.
25	C-T-R 0005	TUBOS RIVAL	0,00	49,00	H	0,00	0,00	8,00	MATERIAL DE C
26	C-B-001	BLOQUE	0,00	0,00	H	0,00	0,00	12,00	MATERIALES D.
27	E-T-003	TUBOS FLUORE...	0,00	0,00	H	0,00	10,00	0,00	MATERIAL DE C
28	M-T-F. 0001	TUBOS FLUORE...	12,00	35,00	H	10,00	10,00	12,00	MATERIAL DE E
29	C-T-F. 40W	TUBOS FLUORE...	0,00	0,00	H	10,00	0,00	0,00	MATERIAL DE E

Figura 82

Así queda con los datos ingresados, para mostrar el reporte damos clic en Generar Reporte (Ver Figura 83), y muestra los resultado requeridos. (Ver Figura 84).

The 'Kardex' window now shows the selected material: 'TUBO PLASTIGAMA ROSC PVC DE 1/2'. The table below displays the details for this material:

Código	ROSC0001	Stock Mínimo	0,00	Existencias	80,00	Vr. Compra	0,00
Grupo	INSUMOS	SubGrupo	CARPINTERO	Marca	PLASTIGAMA	Fecha Ult. Cons.	

The 'Generar Reporte' button is still present on the right.

Figura 83

Universidad Técnica Estatal de Quevedo
 Unidad de Servicios Generales (USG)
 1201143714
 Campus Manuel Haza Alvarez, KM 1 1/2 vía a Santo Domingo de los Tsáchilas
 05287878
 Quevedo - Los Ríos - Ecuador
 compraspublicas@uteq.edu.ec | www.uteq.edu.ec

Movimientos del Material / Insumo: TUBO PLASTIGAMA ROSC PVC DE 1/2

Id Kardex	Centro Almacenamiento	Fecha Registro	Concepto	Entradas	Salidas	Saldo	Precio	Nota
24		02/06/2013 14:03:00	+K Inventario Inicial	100,00	0,00	0,00	0,00	SASA

Página: 1

Figura 84