

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

PROYECTO PREVIO APROBACIÓN PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS COMPUTACIONALES

TEMA:

“Sistema de Gestión de la Investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN”

APLICATIVO:

“Sistema Informático para la Gestión de la Investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN”

AUTOR: HERRERA MORALES CRISTIAN PATRICIO

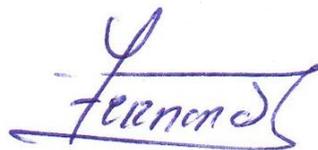
DIRECTOR: ING. MSC. JOSÉ FERNANDO GARRIDO SÁNCHEZ

Ibarra – Ecuador

2013

CERTIFICACIÓN

Certifico que la Tesis “**SISTEMA DE GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA PARA EL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA DE LA UTN**” con el aplicativo “**DESARROLLO E IMPLEMENTACION DEL SISTEMA INFORMÁTICO PARA LA GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA PARA EL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA DE LA UTN**” ha sido realizada en su totalidad por el señor: **HERRERA MORALES CRISTIAN PATRICIO** portador de la cédula de identidad número: **1002957916**.



Ing. Msc. José Fernando Garrido Sánchez

DIRECTOR DE TESIS

CERTIFICACIÓN

Ibarra, 15 de Julio de 2013

Señores

UNIVERSIDAD TÉCNICA DEL NORTE

Presente

De mis consideraciones.

Siendo auspiciantes del proyecto de tesis del Egresado **HERRERA MORALES CRISTIAN PATRICIO** con CI: **1002957916** quien desarrolló su trabajo con el tema “**SISTEMA DE GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA PARA EL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA DE LA UTN**” con el aplicativo “**SISTEMA INFORMÁTICO PARA LA GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA PARA EL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA DE LA UTN**”, me es grato informar que se han superado con satisfacción las pruebas técnicas y la revisión de cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado y realizado por parte del egresado **HERRERA MORALES CRISTIAN PATRICIO**. Una vez que hemos recibido la capacitación y documentación respectiva, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de nuestra empresa/institución.

El egresado **HERRERA MORALES CRISTIAN PATRICIO** puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente



Dr. Marcelo Dávalos
DIRECTOR DEL CUICYT



UNIVERSIDAD TÉCNICA DEL NORTE

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE INVESTIGACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, **HERRERA MORALES CRISTIAN PATRICIO**, con cedula de identidad Nro. **1002957916**, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4, 5 y 6, en calidad de autor del trabajo de grado denominado: “**SISTEMA DE GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA DEL CUICYT-UTN**”, que ha sido desarrollada para optar por el título de Ingeniería en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte

HERRERA MORALES CRISTIAN PATRICIO

10029579165

Ibarra 15 de Julio del 2013



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA
AUTORIZACIÓN DE USO Y PUBLICACIÓN
A FAVOR DE LA
UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual ponemos a disposición la siguiente investigación:

DATOS DE CONTACTO	
CEDULA DE IDENTIDAD	1002957916
APELLIDOS Y NOMBRES	HERRERA MORALES CRISTIAN PATRICIO
DIRECCIÓN	OTAVALO – SAN RAFAEL – CALLE LA ESTACIÓN
EMAIL	cristianpatricio_hm@hotmail.com
TELÉFONO FIJO	062918724
TELÉFONO MÓVIL	0991078477

DATOS DE LA OBRA	
TITULO	“SISTEMA DE GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA PARA EL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA DE LA UTN”
AUTOR	HERRERA MORALES CRISTIAN PATRICIO
FECHA	15 DE JULIO DEL 2013
PROGRAMA	PREGRADO
TITULO OPTADO	INGENIERÍA EN SISTEMAS COMPUTACIONALES
DIRECTOR	ING. MSC. GARRIDO SÁNCHEZ JOSÉ FERNANDO

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, HERRERA MORALES CRISTIAN PATRICIO, con cedula de identidad Nro. 1002957916, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la biblioteca de la universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 143.



HERRERA MORALES CRISTIAN PATRICIO

1002957916

Ibarra 15 de Julio del 2013

DEDICATORIA

Dedico este trabajo a Dios por su infinito amor y misericordia, porque siempre puedo contar con él en los buenos y malos momentos, permitiéndome levantar en las adversidades, dándome fuerzas para seguir adelante y por haberme concedido inteligencia, paciencia, madurez y amor por la carrera que elegí para mi vida profesional.

A mis padres Manuel Herrera y Victoria Morales por haberme dado la vida, por inculcarme buenos valores y por el apoyo inconstante que me brindaron; en especial a mi madre quien fue mi fuente de apoyo y de inspiración, brindándome a cada momento palabras de aliento en los malos momentos.

A mis hermanos William, Jacqueline, Milton, Maritza, Rodrigo, Wilson, Carlos y en especial a mi Hermana Elizabeth quien ha sido como una segunda madre para mí, brindándome todo su apoyo y dándome fuerzas para que siga adelante; y de forma infinita se lo dedico a mi hermano Franklin que aunque ya no esté en este mundo, me motivó a salir adelante enfrentando las adversidades siempre con una sonrisa en el rostro.

A mis sobrinos Richard y Mateo ya que son el regalo más hermoso con el que Dios bendice mi vida y mi hogar.

A mí mismo que nunca me di por vencido, que aunque haya pasado un sinnúmero de situaciones supe levantarme para poder llegar al final del camino.

AGRADECIMIENTO

A través de estas líneas quiero dar las gracias a las personas que en forma directa o indirecta me brindaron su ayuda en la elaboración de este trabajo de grado.

Al equipo humano del Departamento de Informática que me permitió ser parte de la gran revolución tecnológica que están impulsando, por brindarme su apoyo, conocimientos y amistad.

Al Ingeniero Fernando Garrido, mi tutor de tesis que siempre me ayudó con sus consejos y valiosos conocimientos para que este trabajo de grado se llegue a terminar con éxito.

Al equipo humano del Instituto de postgrado quienes me abrieron las puertas para que pueda desarrollar nuevos conocimientos.

A la Ingeniera Viviana Burgos que siempre estuvo presta a brindar su experiencia profesional, quien a más de ser una excelente profesional fue una excelente amiga.

Al equipo humano del CUICYT quienes me recibieron de la mejor manera, dándome todas las facilidades para que pueda culminar mi trabajo de grado.

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN	ii
DEDICATORIA	vii
AGRADECIMIENTO.....	viii
RESUMEN.....	xxi
SUMMARY	xxii
INTRODUCCIÓN	1
1. ANTECEDENTES.....	2
2. PROBLEMA	6
3. OBJETIVOS.....	7
3.1. OBJETIVO GENERAL	7
3.2. OBJETIVOS ESPECÍFICOS	7
4. ALCANCE.....	8
5. JUSTIFICACIÓN.....	9
CAPÍTULO I.....	10
1.1 ORACLE DEVELOPER SUITE 10G	11
1.1.1 Forms Developer	11
1.1.2 Reports Developer.....	21
1.1.3 Designer	23
1.2 LENGUAJE PL/SQL.....	27
1.2.1 Características y Ventajas	27
1.2.2 Herramientas y Clientes	28
1.2.3 Sintaxis Bloque PL/SQL.....	29
1.2.4 Declaraciones	29
1.2.5 Elementos del Lenguaje	32
1.2.6 Manejo de Cursores.....	34
1.2.7 Manejo de Excepciones.....	37
1.2.8 Procedimientos	38
1.2.9 Funciones	39
1.2.10 Borrar Procedimientos y Funciones	40
1.2.11 Parámetros Válidos	40
1.2.12 Disparadores (triggers).....	40
1.3 ORACLE 10G.....	44
1.3.1 RDBMS.....	44
1.3.2 Diccionario de Datos.....	45
1.3.3 Conexión a un Sistema Gestor de Base de Datos.....	45

1.3.4	Herramientas de los RDBMS	47
1.3.5	Base de Datos Objeto-Relacional.....	47
1.3.6	Mejoras del Enfoque Orientado a Objetos	48
1.3.7	Historia de Oracle.....	48
1.3.8	Características	50
1.3.9	Requerimientos de Hardware y Software.....	50
1.3.10	Otros Servidores.....	51
1.3.11	Estrategia de Oracle	51
1.3.12	Servidor Oracle	51
1.3.13	Elementos del Servidor Oracle.....	52
1.3.14	Conexiones.....	52
1.3.15	Estructura de la Base de Datos Oracle	52
1.3.16	Instancia de la Base de Datos	53
1.3.17	Procesamiento de Instrucciones SQL.....	55
1.3.18	Almacenamiento.....	56
1.3.19	Transacciones	57
1.3.20	Usuarios.....	58
1.3.21	Privilegios.....	60
1.3.22	Otorgar/Quitar Privilegios.....	61
1.3.23	Roles.....	62
1.3.24	Pérdida de Información	64
1.3.25	Copias de Seguridad (Backups)	65
1.3.26	Herramientas de Oracle	66
CAPÍTULO II		68
2.1	DOCUMENTO DE VISIÓN	69
2.1.1	Propósito.....	69
2.1.2	Posicionamiento	70
2.1.3	Descripción de los Interesados y Usuarios.....	71
2.1.4	Perfiles de los Stakeholders.....	73
2.1.5	Necesidades de los Interesados y Usuarios	76
2.1.6	Vista General del Producto.....	77
2.1.7	Características del Producto	80
2.2	PLAN DE DESARROLLO DE SOFTWARE.....	81
2.2.1	Propósito.....	81
2.2.2	Alcance.....	81
2.2.3	Vista General del Proyecto.....	82
2.2.4	Organización del Proyecto	87

2.2.5	Plan del Proyecto.....	88
2.2.6	Seguimiento y Control del Proyecto	92
2.3	ESPECIFICACIÓN DE CASOS DE USO	94
2.3.1	Especificación Caso de Uso: Administrador del Sistema.	94
2.3.2	Especificación Caso de Uso: Investigadores.....	95
2.3.3	Especificación Caso de Uso: Convocatorias	96
2.3.4	Especificación Caso de Uso: Presentación de Proyectos	98
2.3.5	Especificación Caso de Uso: Evaluación de Proyectos.....	99
2.3.6	Especificación Caso de Uso: Aprobación de Proyectos.....	100
2.3.7	Especificación Caso de Uso: Oficialización de Proyectos	101
2.3.8	Especificación Caso de Uso: Ejecución de Proyectos.....	102
2.3.9	Especificación Caso de Uso: Cierre de Proyectos.....	103
2.3.10	Especificación Caso de Uso: Eventos	104
2.3.11	Especificación Caso de Uso: Publicaciones	105
2.4	VISTA LÓGICA	107
2.4.1	Modelo Entidad Relación.....	107
2.5	VISTA DE IMPLEMENTACIÓN.....	108
2.5.1	Diagramas de Actividades.....	108
2.5.2	Diagramas de Componentes.....	115
2.5.3	Diagrama de Arquitectura	115
2.6	CASOS DE PRUEBA.....	116
2.6.1	Especificación de Caso de Prueba: Ingresar Investigador.....	116
2.6.2	Especificación de Caso de Prueba: Ingresar Currículo	117
2.6.3	Especificación de Caso de Prueba: Crear Convocatoria	118
2.6.4	Especificación de Caso de Prueba: Asignar Evaluadores	119
2.6.5	Especificación de Caso de Prueba: Asignar Proyectos	120
2.6.6	Especificación de Caso de Prueba: Registrar Proyecto.....	121
2.6.7	Especificación de Caso de Prueba: Evaluar Proyecto	122
2.6.8	Especificación de Caso de Prueba: Aprobar Proyecto	123
2.6.9	Especificación de Caso de Prueba: Priorizar Proyecto.....	124
2.6.10	Especificación de Caso de Prueba: Cargar Proyectos POA	125
2.6.11	Especificación de Caso de Prueba: Crear Tipo Informe	127
2.7	LISTA DE RIESGOS.....	128
2.7.1	Descripción de los Riesgos	128
CAPÍTULO III.....		132
2.8	CONCLUSIONES	133
2.9	RECOMENDACIONES	133

GLOSARIO DE TÉRMINOS	134
BIBLIOGRAFÍA.....	137
ANEXOS.....	141
A. ANTEPROYECTO DE TESIS	142
B. DICCIONARIO DE DATOS	154
B.1. NOMBRE DE LAS TABLAS DEL SISTEMA.....	154
B.2. EXPLICACIÓN DE LOS CAMPOS DE CADA TABLA	157
C. GUÍA DE PROGRAMACIÓN	186
C.1. ESTÁNDARES DE PROGRAMACIÓN	186
C.2. ESTANDARIZACIÓN DEL DISEÑO DE LA BASE DE DATOS	187
C.3. ESTANDARIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN	188
D. MANUAL DE INSTALACIÓN	196
D.1. ORACLE ® 10G DATABASE SERVER	196
D.2. ORACLE ® 10G DEVELOPER SUITE	198
D.3. ORACLE ® 10G APPLICATION SERVER	199
E. PROTOTIPO DE INTERFAZ DE USUARIO	203
E.1. ARCHIVOS DE CONFIGURACIÓN.....	203
E.2. PERSONALIZACIÓN DE LA PÁGINA PRINCIPAL DE LA APLICACIÓN.....	204
E.3. FUNCIONES Y PROCEDIMIENTOS PARA EJECUTAR LA PLANTILLA	206
F. MANUAL DE USUARIO	214
F.1. INGRESO AL SISTEMA INFORMÁTICO INTEGRADO UTN	215
F.2. INGRESO AL SISTEMA PARA LA GESTIÓN DE LA INVESTIGACIÓN.....	215
F.2.1. Sección Parámetro.....	216
F.2.2. Sección Mantenimiento.....	231
F.2.3. Sección Procesos	242
F.2.4. Sección Reportes	246
G. REGLAMENTO INTERNO DEL CUICYT	248

ÍNDICE DE FIGURAS

Figura 1. Áreas de control CUICYT.....	5
Figura 2. Alcance del Sistema.....	8
Figura 1.1. Navegador de Objetos.....	13
Figura 1.2. Módulo de Formularios.....	13
Figura 1.3. Módulo de Menús.....	17
Figura 1.4. Módulo de Biblioteca PL/SQL.....	18
Figura 1.5. Paleta de Propiedades.....	19
Figura 1.6. Editor de Diseño.....	20
Figura 1.7. Estructura Navegador de Objetos de Reports.....	21
Figura 1.8. Modelo de Datos.....	21
Figura 1.9. Disposición de Papel.....	22
Figura 1.10. Pantalla de Parámetros de Papel.....	23
Figura 1.11. Pantalla Componentes de Oracle 10g Designer.....	27
Figura 1.12. Herramientas y Clientes.....	29
Figura 1.13. Conexión Local.....	46
Figura 1.14. Cliente/Servidor.....	46
Figura 1.15. Instancia de la Base de Datos.....	53
Figura 1.16. Almacenamiento.....	56
Figura 1.17. Estructura Lógica.....	57
Figura 1.18. Privilegios de Usuario.....	58
Figura 2.1. Perspectiva del Producto.....	78
Figura 2.2. Descripción del Proceso Unificado.....	90
Figura 2.3. Caso de Uso Administrador del Sistema.....	94
Figura 2.4. Caso de Uso Investigadores.....	95
Figura 2.5. Caso de Uso Convocatorias.....	96
Figura 2.6. Caso de Uso Presentación de Proyectos.....	98
Figura 2.7. Caso de Uso Evaluación de Proyectos.....	99
Figura 2.8. Caso de Uso Aprobación de Proyectos.....	100
Figura 2.9. Caso de Uso Oficialización de Proyectos.....	101
Figura 2.10. Caso de Uso Ejecución de Proyectos.....	102
Figura 2.11. Caso de Uso Cierre de Proyectos.....	103
Figura 2.12. Caso de Uso Eventos.....	104
Figura 2.13. Caso de Uso Publicaciones.....	105
Figura 2.14. Modelo Entidad Relación.....	107
Figura 2.15. Diagrama de Actividad Crear Usuarios.....	108

Figura 2.16. Diagrama de Actividad Crear Roles.	108
Figura 2.17. Diagrama de Actividad Ingresar Investigador.	109
Figura 2.18. Diagrama de Actividad Ingresar Horario de Trabajo.	109
Figura 2.19. Diagrama de Actividad Ingresar Currículo.	110
Figura 2.20. Diagrama de Actividad Crear Convocatoria.	110
Figura 2.21. Diagrama de Actividad Ingresar Procedimiento.	111
Figura 2.22. Diagrama de Actividad Asignar Evaluadores.	111
Figura 2.23. Diagrama de Actividad Asignar Proyectos.	112
Figura 2.24. Diagrama de Actividad Registrar Proyecto.	112
Figura 2.25. Diagrama de Actividad Evaluar Proyecto.	113
Figura 2.26. Diagrama de Actividad Aprobar Proyecto.	113
Figura 2.27. Diagrama de Actividad Oficializar Proyecto.	114
Figura 2.28. Diagrama de Actividad Ejecutar y Cerrar Proyecto.	114
Figura 2.29. Diagrama de Componentes.	115
Figura 2.30. Diagrama de Arquitectura.	115
Figura A.4.1. Alcance del Proyecto.	146
Figura A.4.2. Visión General del Proyecto.	146
Figura A.4.3. Arquitectura.	148
Figura A.4.4. Fases Metodología de Desarrollo.	149
Figura A.4.5. Metodología RUP.	149
Figura F.1. Barra de Herramientas.	214
Figura F.2. Pantalla de Ingreso al Sistema Integrado.	215
Figura F.3. Pantalla del Sistema Integrado de la UTN.	215
Figura F.4. Pantalla del Sistema de Gestión de la Investigación.	216
Figura F.5. Pantalla Parámetros del Sistema.	216
Figura F.6. Pantalla Áreas de Investigación.	217
Figura F.7. Pantalla Cobertura Proyecto.	217
Figura F.8. Pantalla Definiciones de Investigación.	218
Figura F.9. Pantalla Cruce de Ejes Transversales.	218
Figura F.10. Pantalla Líneas de Investigación.	219
Figura F.11. Pantalla Políticas de Investigación.	219
Figura F.12. Pantalla Categoría Proyecto.	220
Figura F.13. Pantalla Ejes Transversales.	220
Figura F.14. Pantalla Estados Informe.	221
Figura F.15. Pantalla Estados Proyecto.	221
Figura F.16. Pantalla Vigencia Formato.	222
Figura F.17. Pantalla Estructura Formato.	222

Figura F.18. Pantalla Niveles de Participación.....	223
Figura F.19. Pantalla Parámetros de Evaluación.....	224
Figura F.20. Pantalla Períodos Proyecto.....	224
Figura F.21. Pantalla Porcentajes de Evaluación.....	225
Figura F.22. Pantalla Sectores de Impacto.....	225
Figura F.23. Pantalla Tipo Convocatoria.....	226
Figura F.24. Pantalla Tipo Documento Respaldo.....	226
Figura F.25. Pantalla Tipo Evento.....	227
Figura F.26. Pantalla Tipo Formato Proyecto.....	227
Figura F.27. Pantalla Tipo Informe Proyecto.....	228
Figura F.28. Pantalla Tipo Objetivo.....	228
Figura F.29. Pantalla Tipo Participación.....	229
Figura F.30. Pantalla Tipo Proyecto.....	229
Figura F.31. Pantalla Tipo Rubro.....	230
Figura F.32. Pantalla Tipología Proyecto.....	230
Figura F.33. Pantalla Ingresar Investigador.....	231
Figura F.34. Pantalla Currículo Investigador.....	231
Figura F.35. Pantalla Convocatorias.....	232
Figura F.36. Pantalla Procedimiento Convocatorias.....	233
Figura F.37. Pantalla Asignar Evaluador Convocatoria.....	234
Figura F.38. Pantalla Asignar Proyecto Evaluador.....	235
Figura F.39. Pantalla Ingresar Semillero.....	236
Figura F.40. Pantalla Evaluar Semillero.....	236
Figura F.41. Pantalla Aprobar Semillero.....	237
Figura F.42. Pantalla Aprobar Semillero.....	238
Figura F.43. Pantalla Administrar Evento.....	238
Figura F.44. Pantalla Facilitadores Evento.....	239
Figura F.45. Pantalla Asistentes Evento.....	240
Figura F.46. Pantalla Contenido Evento.....	241
Figura F.47. Pantalla Artículos Científicos.....	241
Figura F.48. Pantalla Producción Científica.....	242
Figura F.49. Pantalla Ingresar Proyecto.....	243
Figura F.50. Pantalla Evaluar Proyecto.....	244
Figura F.51. Pantalla Aprobar Proyecto.....	244
Figura F.52. Pantalla Oficializar Proyecto.....	245
Figura F.53. Pantalla Ejecutar Proyecto.....	246
Figura F.54. Pantalla Proyecto por Periodo.....	246

Figura F.55. Pantalla Proyecto por Área de Investigación.	247
Figura F.56. Pantalla Proyecto Avance Técnico Económico.....	247

ÍNDICE DE TABLAS

Tabla 1.1. Elementos del Editor de Diseño.....	20
Tabla 1.2. Excepciones del Sistema.....	38
Tabla 1.3. Requerimientos de Hardware y Software.....	50
Tabla 1.4. Privilegios de Usuario.....	59
Tabla 1.5. Privilegios de Objeto.....	61
Tabla 2.1. Definición del Problema.....	70
Tabla 2.2. Definición de la Posición del Producto.....	71
Tabla 2.3. Descripción de los interesados.....	72
Tabla 2.4. Resumen de Usuarios.....	72
Tabla 2.5. Perfil del Coordinador del Proyecto.....	73
Tabla 2.6. Perfil Responsable del Proyecto.....	74
Tabla 2.7. Perfil Responsable del Proyecto.....	74
Tabla 2.8. Perfil Responsable del desarrollo del Sistema.....	75
Tabla 2.9. Perfil de Usuarios.....	75
Tabla 2.10. Necesidades de los interesados y Usuarios.....	77
Tabla 2.11. Resumen de Capacidades.....	78
Tabla 2.12. Costos y Precios.....	79
Tabla 2.13. Roles y Responsabilidades.....	88
Tabla 2.14. Plan de Fases.....	88
Tabla 2.15. Plan de Fases: Itos.....	89
Tabla 2.16. Calendario de la Fase de Inicio.....	91
Tabla 2.17. Calendario de la Fase de Elaboración.....	92
Tabla A.1. Presupuesto.....	150
Tabla A.2. Temas Afines.....	151
Tabla A.3. Cronograma de Actividades.....	153
Tabla B.1. Atributos de la tabla MGI_TAB_ACTIVIDAD_EJECUTADA.....	157
Tabla B.2. Atributos de la tabla MGI_TAB_AREA_INVE_PROYECTO.....	157
Tabla B.3. Atributos de la tabla MGI_TAB_AREA_INVESTIGACION.....	157
Tabla B.4. Atributos de la tabla MGI_TAB_AREINVE_LININV_EJETRA.....	158
Tabla B.5. Atributos de la tabla MGI_TAB_ARTICULOS_CIENTIFICOS.....	158
Tabla B.6. Atributos de la tabla MGI_TAB_ASISTENTES.....	158
Tabla B.7. Atributos de la tabla MGI_TAB_AVANCE_OBJETIVOS.....	159
Tabla B.8. Atributos de la tabla MGI_TAB_BENEFICIARIOS.....	159
Tabla B.9. Atributos de la tabla MGI_TAB_CATEGORIA_PROYECTO.....	159

Tabla B.10. Atributos de la tabla MGI_TAB_CONCLUSIONES_Y_RECOMEN.	160
Tabla B.11. Atributos de la tabla MGI_TAB_CONTENIDO_EVENTO.	160
Tabla B.12. Atributos de la tabla MGI_TAB_CONV_AREA_INV.	160
Tabla B.13. Atributos de la tabla MGI_TAB_CONV_LIN_INV.	161
Tabla B.14. Atributos de la tabla MGI_TAB_CONV_POL_INV.	161
Tabla B.15. Atributos de la tabla MGI_TAB_CONVOCATORIA.	161
Tabla B.16. Atributos de la tabla MGI_TAB_CURSOS.	162
Tabla B.17. Atributos de la tabla MGI_TAB_CURSOS.	162
Tabla B.18. Atributos de la tabla MGI_TAB_DOCUMENTOS_RESPALDO.	162
Tabla B.19. Atributos de la tabla MGI_TAB_EJES_TRANSVERSALES.	163
Tabla B.20. Atributos de la tabla MGI_TAB_EQUIPO_TRABAJO.	163
Tabla B.21. Atributos de la tabla MGI_TAB_ESTADO_INFORME.	163
Tabla B.22. Atributos de la tabla MGI_TAB_ESTADO_PROYECTO.	164
Tabla B.23. Atributos de la tabla MGI_TAB_ESTRUCTURA_FORMATO.	164
Tabla B.24. Atributos de la tabla MGI_TAB_EVALUACION_PROYECTO.	164
Tabla B.25. Atributos de la tabla MGI_TAB_EVALUADORES.	165
Tabla B.26. Atributos de la tabla MGI_TAB_EVENTOS.	166
Tabla B.27. Atributos de la tabla MGI_TAB_EXPERIENCIA_INVESTIGAC.	166
Tabla B.28. Atributos de la tabla MGI_TAB_EXPOSICIONES_PRESENTAD.	166
Tabla B.29. Atributos de la tabla MGI_TAB_FACILITADORES.	167
Tabla B.30. Atributos de la tabla MGI_TAB_FORMACION_ACADEMICA.	167
Tabla B.31. Atributos de la tabla MGI_TAB_FUENTE_VERIFICACION.	167
Tabla B.32. Atributos de la tabla MGI_TAB_HORARIO_TRABAJO.	168
Tabla B.33. Atributos de la tabla MGI_TAB_IDICADOR.	168
Tabla B.34. Atributos de la tabla MGI_TAB_IDICADOR.	169
Tabla B.35. Atributos de la tabla MGI_TAB_INSTITUCION_ASOCIADA.	169
Tabla B.36. Atributos de la tabla MGI_TAB_INVESTIGADORES.	169
Tabla B.37. Atributos de la tabla MGI_TAB_LINEA_BASE.	170
Tabla B.38. Atributos de la tabla MGI_TAB_LINEA_INVE_PROYECTO.	170
Tabla B.39. Atributos de la tabla MGI_TAB_LINEA_INVESTIGACION.	170
Tabla B.40. Atributos de la tabla MGI_TAB_MARCO_LOGICO.	171
Tabla B.41. Atributos de la tabla MGI_TAB_MES_PERIODO.	171
Tabla B.42. Atributos de la tabla MGI_TAB_NIVEL_PARTICIPACION.	172
Tabla B.43. Atributos de la tabla MGI_TAB_OBSERVACIONES.	172
Tabla B.44. Atributos de la tabla MGI_TAB_PARAMETROS_EVALUACION.	172
Tabla B.45. Atributos de la tabla MGI_TAB_PARAMETROS_PROYECTO.	172
Tabla B.46. Atributos de la tabla MGI_TAB_PATENTES.	173

Tabla B.47. Atributos de la tabla MGI_TAB_PERFIL_PROYECTO.....	173
Tabla B.48. Atributos de la tabla MGI_TAB_PERIODO.....	173
Tabla B.49. Atributos de la tabla MGI_TAB_PERIODO_PROYECTO.....	174
Tabla B.50. Atributos de la tabla MGI_TAB_POL_INVESTIGACION.	174
Tabla B.51. Atributos de la tabla MGI_TAB_PORCENTAJE_EVALUACION.....	174
Tabla B.52. Atributos de la tabla MGI_TAB_PROBLEMAS.....	175
Tabla B.53. Atributos de la tabla MGI_TAB_PROCEDIMIENTO_CONV.....	175
Tabla B.54. Atributos de la tabla MGI_TAB_PRODUCION_CIENTIFICA.....	175
Tabla B.55. Atributos de la tabla MGI_TAB_PROMEDIO_PARÁMETRO.	176
Tabla B.56. Atributos de la tabla MGI_TAB_PROYECTO_EVALUADOR.....	176
Tabla B.57. Atributos de la tabla MGI_TAB_PROYECTOS.....	178
Tabla B.58. Atributos de la tabla MGI_TAB_PROYECTOS.....	178
Tabla B.59. Atributos de la tabla MGI_TAB_PUBLICACIONES.	179
Tabla B.60. Atributos de la tabla MGI_TAB_RESULTADOS.....	179
Tabla B.61. Atributos de la tabla MGI_TAB_RUBRO_MES.....	179
Tabla B.62. Atributos de la tabla MGI_TAB_RUBRO_PERIODO.....	180
Tabla B.63. Atributos de la tabla MGI_TAB_RUBRO_PERIODO_DET.....	180
Tabla B.64. Atributos de la tabla MGI_TAB_SECT_IMPA_PROYECTOS.....	180
Tabla B.65. Atributos de la tabla MGI_TAB_SECTOR_IMPACTO.....	181
Tabla B.66. Atributos de la tabla MGI_TAB_SUPUESTOS.	181
Tabla B.67. Atributos de la tabla MGI_TAB_TIPO_CONVOCATORIA.....	181
Tabla B.68. Atributos de la tabla MGI_TAB_TIPO_DOCUMENTO_RESP.....	181
Tabla B.69. Atributos de la tabla MGI_TAB_TIPO_EVENTO.....	182
Tabla B.70. Atributos de la tabla MGI_TAB_TIPO_FORMATO.....	182
Tabla B.71. Atributos de la tabla MGI_TAB_TIPO_INFORME.....	182
Tabla B.72. Atributos de la tabla MGI_TAB_TIPO_LOCALIDAD.....	182
Tabla B.73. Atributos de la tabla MGI_TAB_TIPO_OBJETIVO.....	183
Tabla B.74. Atributos de la tabla MGI_TAB_TIPO_PARTICIPACIOM.....	183
Tabla B.75. Atributos de la tabla MGI_TAB_TIPO_PROYECTO.....	183
Tabla B.76. Atributos de la tabla MGI_TAB_TIPO_RUBRO.....	183
Tabla B.77. Atributos de la tabla MGI_TAB_TIPOLOGIA.....	184
Tabla B.78. Atributos de la tabla MGI_TAB_TRASFERENCIA.....	184
Tabla B.79. Atributos de la tabla MGI_TAB_USUARIOS.....	184
Tabla B.80. Atributos de la tabla MGI_TAB_VIGENCIA_FORMATO.....	185
Tabla C.1. Tipos de Objetos de la Base de Datos.....	187
Tabla C.2. Áreas de Desarrollo.....	187
Tabla C.3. Nombres de los Objetos.....	188

Tabla C.4. Ejemplos de Definición de Objetos.....	188
Tabla C.5. Prefijos de Objetos.	189
Tabla C.6. Variables y Tipos de Datos.....	194

RESUMEN

En una sociedad que basa su economía en la producción, distribución y uso del conocimiento y la información, la Universidad Técnica del Norte se constituye en una organización clave para su desarrollo. No obstante, el conocimiento y la información aportarán ventajas competitivas a la institución siempre que ésta, habiéndose planteado metas y objetivos en relación y en contexto, adopte modelos organizacionales pertinentes e implemente estrategias concretas y cuantificadas que le permitan lograrlos.

La universidad ha desarrollado la función misional de investigación, en el ámbito local desde muchos años atrás con una amplia oferta de formación de profesionales, se ha constituido en la principal línea funcional para las instituciones, las condiciones y tendencias del contexto actual exigen un desarrollo integrado y consolidado de sus tres funciones.

Es innegable la relación histórica entre el sistema científico-tecnológico y la sociedad. Hasta la actualidad el proceso investigativo ha incursionado en todos los ámbitos de formación social y cultural, generando grandes avances científicos que han ayudado al ser humano a construir grandes inventos y por ende a tener un mejor estilo de vida.

El objetivo del presente trabajo es dar un Sistema de Gestión de la Investigación, que permita a la Universidad Técnica del Norte y al Centro Universitario de Investigación Científica y Tecnológico gestionar de mejor manera las actividades investigativas que realizan sus investigadores.

Para gestionar de mejor manera este proceso investigativo se requiere de sistemas de información que permitan mantener un control permanente de la información generada a través del tiempo, permitiendo que investigaciones futuras dispongan de un respaldo permanente de esta información.

El sistema está implantado en una plataforma Web de Oracle Forms 10g, formando parte del sistema integrado de la Universidad Técnica del Norte “ERP-UTN”. La incorporación de este sistema permite administrar significativamente el número de investigadores activos, así como la cantidad de proyectos de investigación ejecutados, el número de publicaciones de corriente principal y la adjudicación de fondos externos e internos.

SUMMARY

In a society that bases its economy on the production, distribution and use of knowledge and information, the university is a key organization for development. However, the knowledge and information will provide competitive advantages to the university where this, having been raised in relation to goals and objectives and in context, relevant organizational models adopt and implement specific strategies and quantified to enable it to achieve them.

The university has developed the missionary role of research in the local area from many years ago and a wide range of professional training has become the leading institution for functional line, conditions and trends in the current context require an integrated development and consolidated its three functions.

There is no denying the historical relationship between science and technology system and society.

To date the research process has moved into all areas of social and cultural formation, generating scientific breakthroughs that have helped human beings to build great inventions and therefore have a better lifestyle.

The aim of this paper is to present an Environmental Research, to enable the Technical University and North University Center for Scientific Research and Technological better manage research activities done by its researchers.

To better manage this investigative process requires information systems that maintain constant control of information generated over time, allowing future research have a permanent backup of this information.

The system is implemented in a platform Web Oracle Forms 10g as part of the integrated North Technical University "ERP-UTN." The incorporation of this system to manage significantly the number of active researchers and the number of research projects implemented the number of mainstream publications and the award of external and internal funds.

INTRODUCCIÓN



**SISTEMA DE GESTIÓN
DE LA
INVESTIGACIÓN
CIENTÍFICA,
TECNOLÓGICA E
INNOVATIVA PARA EL
CUICYT - UTN**

1. ANTECEDENTES

La Universidad Técnica del Norte (UTN¹), es una academia de carácter público, tiene como misión esencial contribuir al desarrollo educativo, científico, tecnológico, socioeconómico y cultural de la región norte del país a través de la generación y difusión del conocimiento. Forma profesionales críticos, creativos, capacitados, humanistas y éticos comprometidos con el cambio social y con la preservación del medio ambiente.

La UTN está ubicada en la provincia de Imbabura, cantón Ibarra, barrio El Olivo. La Universidad cuenta con el Centro Universitario de Investigación Científica y Tecnológica (CUICYT²) como centro encargado de coordinar la gestión de la investigación que se realiza dentro y fuera de la institución.

La universidad a través del CUICYT desea institucionalizar y sistematizar todo el proceso de investigación en ciencia, tecnología e innovación de manera interactiva y multidisciplinaria para propiciar la creación, adaptación, generación y transferencia de tecnología, en la búsqueda de alternativas viables e innovadoras para la solución de problemas prioritarios de los sectores sociales y productivos de la región norte del país.

El CUICYT a más de impulsar proyectos de investigación es un centro generador de investigadores a través de la investigación en el aula, formando grupos permanentes de estudiantes que se organizan en el aula alrededor de proyectos de investigación, con miras a proponer alternativas de solución a los problemas del entorno; realiza eventos tales como exposiciones de investigación, talleres, seminarios, publicación y difusión de artículos científicos.

➤ Generalidades

En la década de los años 70, un importante sector de profesionales que sentían la necesidad de que el norte del país cuente con un Centro de Educación Superior que responda a los requerimientos propios del sector comienzan a dar los primeros pasos para lo que hoy en día constituye la UTN.

Con este propósito, e identificados los requerimientos urgentes para que se den las respuestas a la solución de problemas y necesidades acordes con el avance técnico y científico de la época, se crean las facultades de Ciencias de la Educación, Administración de Empresas, Enfermería e Ingeniería, y se convoca a los bachilleres a que se inscriban.

¹ Universidad Técnica del Norte

² Centro Universitario de Investigación Científica y Tecnológica.

Las expectativas que se generan son de tal magnitud que estas son determinadas por los centenares de aspirantes a las diversas ramas que acuden a recibir clases en locales de establecimientos educativos de enseñanza media de la ciudad de Ibarra, la mayoría facilitados en forma gratuita.

El profesorado básicamente estaba compuesto por profesionales imbabureños, docentes de los diversos colegios que empezaron trabajando por patriotismo más que por el aspecto remunerativo; igual papel cumplían un pequeño grupo de empleados. En la época de Triunvirato³ Militar se hicieron las gestiones para alcanzar la oficialización de Universidad; el proyecto Decreto de creación que ya tenía el asentimiento de los dos Triunviros, finalmente fue roto por el extinto Gral. Guillermo Durán Arcentales, el 8 de agosto de 1979.

La perseverancia de las autoridades de la UTN, nuevamente en el régimen Constitucional se logra reactivar las acciones tendientes a la legalización como Universidad; para ello se realizaron varias gestiones ante la Honorable Cámara de Representantes (HCR⁴) para alcanzar el Decreto de creación, el cual fue aprobado por este organismo del Estado, pero el Presidente Jaime Roldós lo vetó totalmente el 11 de octubre de 1979. Transcurrido un año, se reinició nuevamente la gestión produciéndose una situación similar, el Congreso reafirma el Decreto de creación pero el Presidente Oswaldo Hurtado Larrea por el mes de junio de 1981 aplica un nuevo veto total.

El cuerpo directivo de aquella época necesitaba encontrar alguna salida para lo cual se solicitó en vano que universidades como la Central, la de Portoviejo y otras acogieran como Extensión a la Universidad Técnica del Norte. Este objetivo se hizo realidad gracias a la Universidad Nacional de Loja que acoge inicialmente el pedido el 3 de septiembre de 1981; y se logra oficializar la Extensión Universitaria con resolución del Honorable Consejo Universitario (HCU⁵), el 31 de marzo de 1982.

Pese a la situación geográfica, el HCU envía comisiones para el análisis académico, administrativo, financiero, etc. de la extensión creada. Los innumerables informes obtenidos por las comisiones detectan graves errores en la conducción y con este motivo se releva de las funciones al Subdirector de la Extensión y se encarga esta función al Dr. Antonio Posso Salgado.

En la estructura orgánica funcional de la Universidad Técnica del Norte se encuentra el Centro Universitario de Investigación Científica y Tecnológica que cuenta con las normativas y reglamentos para la coordinación de la investigación en las diferentes unidades académicas, departamentos e institutos.

³ Gobierno ejercido por tres personas, que normalmente están aliadas.

⁴ Honorable Cámara de Representantes.

⁵ Honorable Consejo Universitario.

Luego de un largo proceso de mejoramiento continuo la investigación científica, la innovación y la transferencia de tecnología han sido los pilares fundamentales del proceso de formación de profesionales en la Universidad Técnica del Norte y ha sido un instrumento muy útil en la búsqueda de alternativas de solución a los problemas prioritarios de la sociedad y la región norte del país.

El CUICYT, con su equipo multidisciplinario de docentes investigadores y funcionarios, promueve e impulsa el desarrollo de la investigación universitaria en las diferentes áreas del conocimiento humano.

Constituye un reto importante la ejecución de programas de vinculación entre la UTN y los sectores productivos, donde se lograrán procesos tecnológicos innovadores, además existen variadas opciones para el acceso a los conocimientos modernos y, a las diversas formas de incorporarlos de manera eficiente a nuestro aparato productivo. Esto ha permitido a la empresa especialmente privada abaratar los costos de producción y maximizar su rendimiento, haciéndola más competitiva.

Por lo expuesto anteriormente, constituye un requisito fundamental y determinante, el afianzar y consolidar la investigación en la Universidad Técnica del Norte, en el ámbito regional y nacional.

➤ **Misión del CUICYT**

El Centro Universitario de Investigación Científica y Tecnológica tiene la misión fundamental de institucionalizar y sistematizar la investigación, de manera interactiva y multidisciplinaria para propiciar la creación, adaptación, generación y transferencia de tecnología, en la búsqueda de alternativas viables e innovadoras para la solución de problemas prioritarios de los sectores sociales y productivos de la región norte del país.

➤ **Visión del CUICYT**

El Centro Universitario de Investigación Científica y Tecnológica, se constituirá en el líder de la excelencia en investigación a nivel local, regional y nacional como núcleo transformador de la sociedad, eje articulador, autogestionario, destacado y preparado para contribuir al desarrollo integral de la región norte del país, generando ciencia, tecnología e innovación; interactuando con los sectores sociales y productivos del entorno, priorizando la sostenibilidad y sustentabilidad del ambiente.

➤ Áreas de Control

El Centro Universitario de Investigación Científica y Tecnológica “CUICYT” es un organismo precedido por el Honorable Consejo Universitario “HCU”, por la Comisión de Evaluación Interna y por el Vicerrectorado Académico.

El CUICYT depende exclusivamente del Vicerrectorado Académico, el consejo directivo está conformado por el Vicerrector Académico, el director y tres investigadores.

A cargo de la dirección está el Dr. Marcelo Dávalos Herrera y como secretaria la Lcda. Mónica Jurado.

Cuenta con Coordinadores de Facultad, un área técnica integrada por el Ing. Lenin Escobar como Coordinador de Proyectos, el Econ. Sherman Ruiz como Analista Económico Financiero y el Lcdo. Marcelo Gudiño como Digitador de Artículos Científicos y un grupo de Docentes Investigadores.

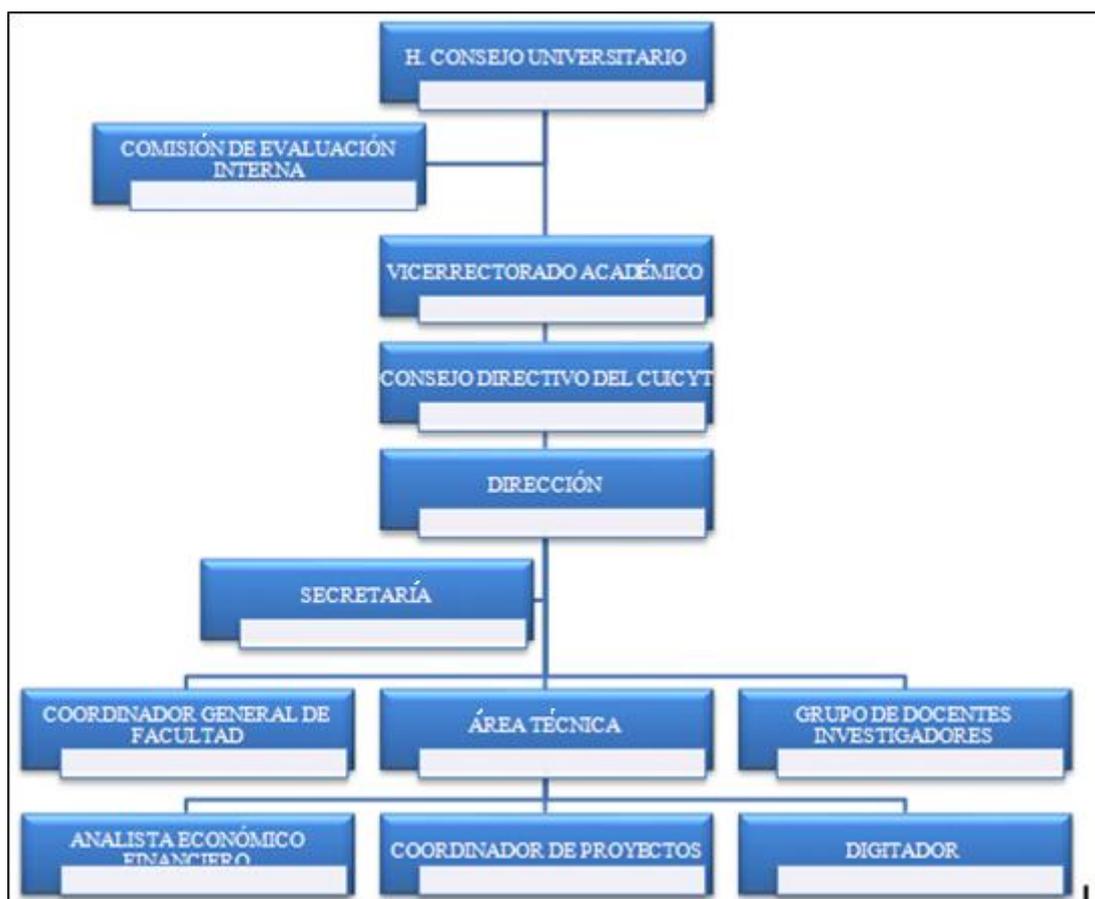


Figura 1. Áreas de control CUICYT.
Fuente: [Propia].

2. PROBLEMA

➤ Situación Actual

En la actualidad el CUICYT realiza el registro de la información (proyectos de investigación, seminarios, talleres, capacitaciones, publicaciones, expo-investigación, etc.) generada en la gestión de la investigación de forma manual; es decir que la documentación de la gestión investigativa que realizan queda archivada en carpetas, lo cual genera muchas limitaciones porque no se tiene un correcto control de esta información y al momento de hacer un levantamiento de la información se emplea mucho tiempo para procesarla ya que la información no está organizada correctamente.

Al no contar con un registro adecuado de la información el centro está limitado a no contar con una fuente de información confiable y oportuna que les permita llevar un control permanente de la gestión investigativa. Todo esto ocasiona que no se pueda hacer un reporte oportuno de cada una de las actividades que realiza el CUICYT.

Esta situación ha limitado que la mayoría de las actividades investigativas que se desarrollan dentro de la universidad y que son gestionadas por el CUICYT no sean conocidas por la comunidad universitaria ni mucho menos por la sociedad.

➤ Prospectiva

De continuar con el registro manual de la información que es generada en la gestión de la investigación; para los próximos años el futuro será incierto, ya que han existido reformas aplicadas por el Gobierno Nacional para el tratamiento de la información con lo referente a la investigación que se realiza dentro de las universidades, es por ello que se hace necesario la implementación del sistema para que por medio del Centro Universitario de Investigación Científica y Tecnológica de la Universidad Técnica del Norte, se pueda tener un control oportuno y así evitar la inconsistencia de información, lo cual podría acarrear pérdidas económicas, pérdida de credibilidad y confianza por parte de la sociedad.

➤ Planteamiento del Problema

El Centro Universitario de Investigación Científica y Tecnológica registra la información generada en los procesos de gestión de la investigación de forma manual, lo cual provoca problemas tales como pérdida e inseguridad de la información, demora en la elaboración de informes y no se cuenta con información consistente que ayude para la toma de decisiones.

3. OBJETIVOS

3.1. OBJETIVO GENERAL

- Diseñar e implementar un sistema de gestión de la investigación científica, tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN que ayude a mantener un control oportuno y permanente de la información científica generada por los investigadores, docentes y estudiantes de la UTN.

3.2. OBJETIVOS ESPECÍFICOS

- Automatizar los procesos de gestión de la investigación que realiza el CUICYT, mediante un análisis profundo de requerimientos.
- Elaborar un sistema informático con las debidas seguridades garantizando la disponibilidad e integridad de la información.
- Desarrollar un sistema informático que sea ágil, amigable y eficiente para mejorar las actividades del quehacer institucional y en especial para el departamento del CUICYT.
- Garantizar la seguridad de la información generada a través del “Sistema de gestión de la Investigación Científica, Tecnológica e Innovativa”, a través de los módulos de gestión de auditoría de base de datos y gestión de seguridades informáticas los mismos que forman parte del sistema integrado de la UTN.
- Investigar y estudiar el uso de herramientas informáticas y normas de desarrollo de software que se utilizan en la Universidad para garantizar la calidad en el desarrollo informático.
- Planificar evaluaciones constantes con los usuarios sobre el funcionamiento del sistema en desarrollo, para verificar el cumplimiento de los requerimientos planteados por el CUICYT.
- Generar información integra, confiable y disponible en cualquier momento.
- Analizar la integración del “Sistema de gestión de la investigación científica, tecnológica e Innovativa” con los sistemas y módulos ya desarrollados en la Universidad para evitar la duplicación de la información.

4. ALCANCE

La implementación del sistema informático beneficia directamente al Centro Universitario de Investigación Científica y Tecnológica, agilizando el trabajo y ofreciendo un mejor servicio a los usuarios, proporcionando en todo momento información confiable, disponible y segura, fundamento principal para optimizar y agilizar la administración de los proyectos de investigación presentados a la dependencia. Además contará con valor agregado en cuanto a la seguridad, rapidez y amigabilidad así como:

- El sistema está diseñado para tener un funcionamiento en modo WEB.
- Funciona a través del sistema integrado de la universidad evitando así el aislamiento total de la comunidad universitaria.
- El aplicativo está montado en el servidor blade de aplicaciones bajo el control y monitoreo de Oracle ® Application Server.
- La base de datos estará montada en el servidor blade que administra el Oracle ® Database Management, por lo que estarán cubiertas las necesidades de hardware para este proyecto.
- El acceso de los usuarios finales será por la red interna de la universidad y a través de Internet.



Figura 2. Alcance del Sistema.
Fuente: [Propia].

5. JUSTIFICACIÓN

Como es de conocimiento la Universidad Técnica del Norte es una entidad legalmente acreditada y por ende está obligada a regirse a las nuevas leyes y políticas de Educación Superior dictaminadas por la Secretaría Nacional de Educación Superior, Ciencia y Tecnología (SENESCYT⁶), quien regula y controla la generación y difusión de conocimientos en las instituciones de educación superior pública y privada.

Dentro de la Universidad Técnica del Norte el Centro Universitario de Investigación Científica y Tecnológica (CUICYT) es la dependencia obligada a generar nuevos conocimientos a través del apoyo económico y técnico brindado a los proyectos de investigación presentados y desarrollados por los docentes e investigadores de la UTN.

En el CUICYT se observó la falta de un sistema informático que les permita coordinar, organizar y agilizar el levantamiento de información. Esta falencia se daba puesto a que existía separación de funciones y por ende la información estaba sometida a diferentes tipos de manipulaciones, dando lugar a fraudes de toda índole.

Al implementar este sistema informático se agiliza el tiempo de respuesta y se garantiza confiabilidad y seguridad en la información.

Es por ello que, para cumplir con las normativas establecidas por la SENESCYT y para mejorar los procesos generados a través de la investigación en el CUICYT, se planteó y se desarrolló el presente proyecto.

⁶ Secretaría Nacional de Educación Superior, Ciencia y Tecnología.

CAPÍTULO I

MARCO TEÓRICO



SISTEMA DE GESTIÓN DE
LA INVESTIGACIÓN
CIENTÍFICA, TECNOLÓGICA
E INNOVATIVA PARA EL
CUICYT - UTN

1.1 ORACLE DEVELOPER SUITE 10G

“Oracle Developer Suite es una suite⁷ de herramientas de desarrollo publicados por la Corporación Oracle”^[W1]. Los componentes empleados para el desarrollo del sistema son:

- Oracle Forms Developer.
- Oracle Reports Developer.
- Oracle Designer.

1.1.1 Forms Developer

“El componente Forms de Developer es la parte del entorno de desarrollo en la que se construyen las pantallas. Proporciona el entorno de trabajo para desarrollar menús, bibliotecas PL/SQL⁸”^[D1], bibliotecas de objetos, paquetes incorporados y objetos de base de datos. Es una herramienta de desarrollo que puede ser usada para crear aplicaciones para introducir, acceder a, modificar o eliminar datos desde una base de datos Oracle en un entorno basado en formularios en línea.

➤ Ciclo de Vida

- Nace en 1985 con la versión 1.0.
- En 1990 lanza la versión 3.0.
- Entre 1993 y 1995 aparece la versión 4.0 y 4.5, con un ambiente gráfico.
- En 1997 la versión 4.5 D1.4w, con la primera versión Web.
- En 1999 lanza la versión 6.0, Developer.
- En 2000 la versión 6i, compatible con la 6.0, java importer, versión en el iAS 9i Release 1.
- En el 2002 lanza la versión 9i –iDS, Java, XML, iAS 9i Release 2.
- En el 2005 lanza la versión 10g, integración con EM, SSO.
- En el 2009 lanza la versión 11g, con WebLogic 11g.

⁷ Conjunto de herramientas de software incluidas en un sólo paquete y que se integran entre sí.

^[W1]WIKIPEDIA.(s.f.).Oracle Developer Suite. Recuperado el 18 de Diciembre del 2012, de http://en.wikipedia.org/wiki/Oracle_Developer_Suite

⁸ PL/SQL es un lenguaje estructurado en bloques.

^[D1]Genero, M. (s.f.). Introducción a Oracle Developer, Recuperado el 28 Octubre del 2012, de <http://es.scribd.com/doc/101126993/Manual-Developer>

➤ **Características**

- Integración con el Application Server Control.
- Soporte para Java 1.4.
- Pre inicialización de runtimes⁹.
- Restricción de parámetros en el URL.
- Mejoras en la integración con el Sistema Operativo.
- Simplificación de la instalación para solo Forms y Reports.
- Integración con Java.
- Mejoras en la herramienta de desarrollo.

➤ **Ventajas**

- Proveer una interface al usuario para insertar, actualizar, eliminar y consultar datos.
- Presentar todo tipo de datos como texto, video, imágenes, el controlador del ActiveX¹⁰, incluyendo JavaBeans¹¹ y Pluggable Java Components.
- Controlar formas a través de diferentes ventanas y transacciones de la Base de Datos.
- Acceder a gráficos y usar menús que integren aplicaciones.
- Pasar datos a un Reports Builder^[D2].

➤ **Entorno de Desarrollo**

Las herramientas primarias usadas para desarrollar formularios personalizados son el navegador de objetos, el editor de diseño y la paleta de propiedades de objetos. Además de los formularios en línea, Oracle Forms es utilizado también para crear y mantener menús de aplicaciones, bibliotecas de unidades de programas, biblioteca de objetos, paquetes incorporados y objetos de base de datos.

⁹ Intervalo de tiempo en el que un programa de computadora se ejecuta en un sistema operativo.

¹⁰ Entorno para definir componentes de software reusables independiente del lenguaje de programación.

¹¹ Modelo de componentes creado por Sun Microsystems para la construcción de aplicaciones en Java.

^[D2] Otiniano, R. (s.f.). Tutorial de Oracle Forms Developer 10g. Recuperado el 3 Junio del 2011, de <http://es.scribd.com/doc/56988520/Oracle-Forms-Developer-10g>

- **Navegador de Objetos**

Los objetos son presentados jerárquicamente tanto los del cliente como los del servidor.

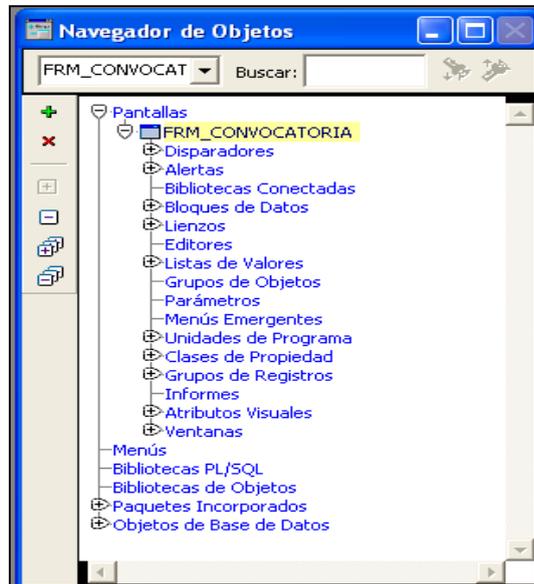


Figura 1.1. Navegador de Objetos.
Fuente: [Propia].

- ❖ **Módulo de Formularios**

El módulo de formularios es el componente principal de las aplicaciones interactivas. También es el módulo más complejo en términos de estructura interna, ya que contiene clases distintas de elementos.



Figura 1.2. Módulo de Formularios.
Fuente: [Propia].

- ✓ **Disparadores:** Un disparador es un bloque de código PL/SQL que se asocia a otro elemento: un formulario, un bloque de datos o un elemento de un bloque de datos. El disparador se “lanza”, o se ejecuta, cuando se producen ciertos eventos.

Los disparadores de una aplicación de formularios contienen el código que se añade a la aplicación, aparte del código que se introduce en las unidades de programa independientes o en bibliotecas.

Los disparadores y su estructura lógica constituyen la mayor parte del diseño de un formulario.

La mayoría de los disparadores que se utilizan son disparadores incorporados, es decir, disparadores proporcionados por el entorno de trabajo de Developer. Cada disparador tiene un nombre específico (por ejemplo, When-Button-Pressed, Post-Query, On-Delete, Key-Help, Pre-Update) que se puede seleccionar de una lista de nombres de disparador.

También se pueden añadir disparadores definidos por el usuario y ejecutarlos de forma explícita.

Los disparadores de teclas, se ejecutan cuando se pulsa una combinación de teclas en la interfaz de usuario. Añadir disparadores de teclas permite reprogramar el teclado para realizar acciones diferentes de las definiciones estándar de las teclas.

- ✓ **Alertas:** Un aviso es un cuadro de diálogo especial que muestra un mensaje con un icono y hasta tres botones, como OK y Cancel, Yes y No, etc. Las alertas pueden ser paradas, de precaución o de información
- ✓ **Bibliotecas Conectadas:** Es un área donde se pueden definir enlaces a librerías o bibliotecas .pll, donde se encuentren las funciones, procedimientos o paquetes.
- ✓ **Bloques de Datos:** El bloque de datos es la unidad de construcción intermedia de los formularios. Un bloque de datos se puede ver de dos formas, como una colección de elementos o como una colección de registros, cada uno de los cuales tiene la misma estructura.

Se puede especificar el número de registros que se van a mostrar a la vez y si se van a mostrar horizontal o verticalmente. Los bloques de datos pueden ser de **datos tabla base** (tabla, vista) y de **control** (conjunto de elementos con valores únicos y un único registro).

El bloque de datos tiene una barra de desplazamiento que permite al usuario gestionar un conjunto de registros mayor que el que se puede mostrar en el canvas. También puede tener un conjunto especial de atributos visuales (elementos programables) que definen un aspecto diferente para el registro en curso, el registro en el que está el cursor.

La navegación dentro del bloque de datos se produce normalmente en el orden en el que se definen los elementos en un registro. El sistema en tiempo de ejecución tiene funciones para moverse de registro en registro, de elemento en elemento y de bloque de datos en bloque de datos.

Cuando se sale del último elemento de un registro, se vuelve normalmente al primer elemento de ese registro. Forms permite volver a definir este comportamiento para poder pasar al siguiente registro (o al registro anterior si se retrocede desde el primer elemento). También permite pasar al bloque de datos siguiente o anterior cuando el movimiento se realiza desde el último o el primer elemento, respectivamente. Se puede especificar qué bloque de datos es el siguiente o el anterior, creando una lista de bloques.

También se le puede decir a Forms que introduzca el nombre de un bloque de datos en el menú de bloques, lo que permite pasar a un bloque de datos eligiendo su nombre en una lista. La función principal de un bloque de datos de tabla base es proporcionar una interfaz a una tabla o, un procedimiento almacenado en la base de datos.

Developer proporciona un asistente de bloques de datos que ayuda a construir bloques de datos a partir del esquema de la base de datos. Este asistente también ayuda a crear bloques de datos maestro-detalle.

- ✓ Los **Elementos**: Son las unidades básicas de desarrollo de formularios. Un elemento tiene muchas propiedades. Cada tipo de elemento tiene sus propiedades. Se puede hacer referencia a los valores de un elemento utilizando un punto como parte de esta sintaxis especial:

<Nombre del bloque de datos>.<nombre del elemento>

Existen diferentes tipos de elementos: Control Active X, Gráfico, Cuadro de verificación, Imagen, Lista, botón, sonido, etc.

Se puede especificar un formato de máscara que establezca el formato de la salida o valide la entrada.

Se puede especificar un valor por defecto para el elemento o especificar el nombre de otro elemento del cual copiar el valor inicial. Se puede especificar una longitud máxima o el rango de datos, etc.

- ✓ Las **Relaciones**: Son un elemento especial que Forms utiliza para estructurar formulario maestro-detalle.

El elemento relación, que pertenece al bloque de datos maestro, expresa la relación del registro maestro con sus registros detallados.

Las propiedades principales de la relación son el nombre del bloque de datos detallado y la condición de unión que Forms utiliza para gestionar la relación.

También se puede especificar algún comportamiento especial respecto a la eliminación de registros maestros (si se tienen que borrar, o no, los registros detallados) o la introducción o actualización de registros detallados cuando no existe un registro maestro.

No todas las formas contienen un único bloque sobre el cual interactuar. Muchas de ellas contienen más de uno relacionados entre sí (maestro detalle). Para crear una relación entre dos bloques se debe situarse sobre el bloque maestro y crear una nueva relación. Seleccionar el bloque detalle en la ventana de creación de relaciones y seleccionar las operaciones de eliminación de registros: en cascada para eliminar los registros hijos junto con el padre, aisladas para no eliminar los registros hijos cuando se elimine el padre, no aisladas para evitar que se borre el registro maestro si hay registros hijos.

- ✓ **Lienzos:** Un lienzo es la base sobre la que se sitúa el texto plano y los elementos. Cada elemento hace referencia a un único lienzo en su hoja de propiedades. Los elementos de un bloque de datos se pueden dividir entre diferentes lienzos.

Un lienzo no es un elemento de interfaz autónomo. Para verlo a él y sus elementos es necesario mostrarlo en una ventana, un área rectangular rodeada por un marco y mantenida por la plataforma GUI.

- ✓ **Ventanas:** La ventana puede tener barras de desplazamiento que permiten moverse a través de los lienzos para ver diferentes vistas, estas pueden ser **documento** (visualiza un elemento documento de la aplicación) y **cuadro de diálogo** (muestra opciones de control operativa de la aplicación).
- ✓ **Menús Emergentes:** Un menú emergente es un menú flotante, que emerge cuando se pulsa con el botón derecho del ratón en un canvas o en un elemento.
- ✓ **Editores:** Es un cuadro de diálogo con un editor de texto que permite la introducción de líneas de texto en un elemento texto. El elemento editor permite especificar el tamaño de la ventana, los atributos visuales, el título del editor y otras propiedades de la ventana. Esto permite la creación de editores con apariencias diferentes para diferentes campos de texto.
- ✓ **Grupo de Registros:** Un grupo de registros es una estructura de datos especial parecida a una tabla con filas y columnas. Un grupo de registros puede ser un grupo de registros procedentes de una consulta o un grupo de registros estáticos. Los grupos de registros se pueden utilizar en LOV, en parámetros de datos que pasan registros a informes o gráficos, o como estructuras de datos PL/SQL.

- ✓ **Lista de Valores (LOV):** Un LOV es un cuadro de diálogo especial que muestra un grupo de registros, permitiendo elegir una fila del grupo, devolviendo un único valor. Un LOV se utiliza como una forma de elegir un conjunto específico de valores.

❖ **Módulo de Menús:**

El módulo de menús es mucho más sencillo que el módulo de formularios.

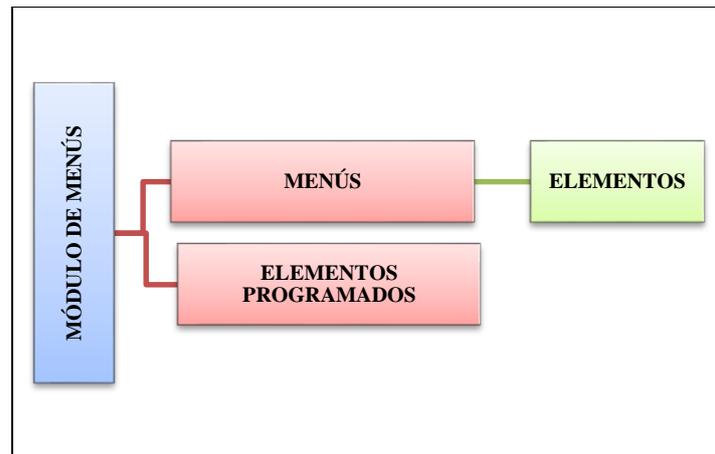


Figura 1.3. Módulo de Menús.

Fuente: [Propia].

Consiste en un conjunto de elementos programables y un conjunto de menús. A su vez cada menú consiste en un conjunto de opciones.

Las opciones pueden ser sencillas o tener uno de los siguientes formatos:

- ✓ **Verificación:** Una opción con una marca de verificación al lado; permiten activar o desactivar opciones a través de menú en vez de mediante cuadros de diálogo.
- ✓ **Opción:** Una opción que pertenece a un grupo de opciones mutuamente exclusivas; si se selecciona una, se deja de seleccionar las otras.
- ✓ **Separador:** Una opción que no hace nada, encargada de separar otras opciones, generalmente mediante un espacio o una línea.
- ✓ **Mágica:** Una opción especial de una plataforma, como cortar, copiar, pegar, deshacer o ayuda.

Cada opción tiene una orden que Developer ejecuta cuando se selecciona, esta puede ser:

- ✓ **Null:** No hace nada, un separador debe tener una orden null.
- ✓ **Menú:** El menú muestra un submenú.
- ✓ **PL/SQL:** El menú ejecuta un bloque PL/SQL.

Una opción también se puede asociar a uno o más papeles de seguridad.

❖ **Módulo de Bibliotecas PL/SQL**

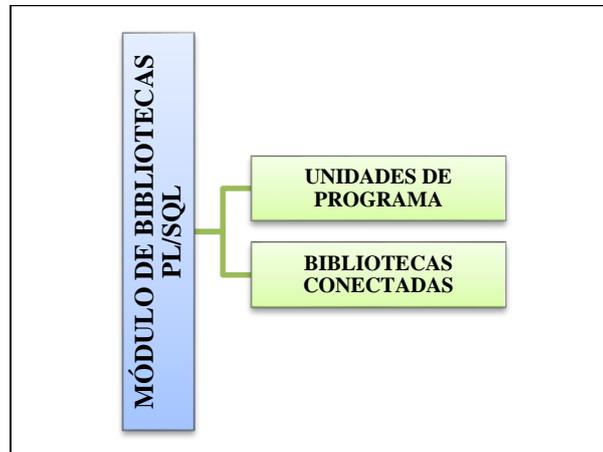


Figura 1.4. Módulo de Biblioteca PL/SQL.
Fuente: [Propia].

Las bibliotecas PL/SQL consisten en un repositorio que sirve para centralizar el código en PL/SQL al que acceden otros tipos de módulos de Oracle Forms. Las bibliotecas se pueden almacenar en el sistema de archivos o en la base de datos como cualquier otro módulo.

Consiste de un conjunto de unidades de programa (procedimientos, funciones, especificación del paquete y el cuerpo del paquete) y un conjunto de bibliotecas asociadas.

❖ **Módulo de Paquetes Incorporados**

Proporcionan una lista de herramientas de manipulación de módulos y otros elementos para Forms, Reports y Graphics. Cada paquete contiene una especificación que lista los subprogramas del paquete junto con su especificación (parámetros y valores devueltos por las funciones).

❖ **Objetos de Base de Datos**

Todos los Builes permiten acceder a los elementos de la base de datos a la que se conecte. Listan todos los usuarios de la base de datos. Para cada usuario, se pueden ver las unidades de programa, bibliotecas, tablas y vistas almacenadas accesibles. Sólo se ven las que se pueden acceder a través del nombre de usuario con el que se ha iniciado una sesión.

• **Paleta de Propiedades**

Permite realizar una búsqueda más rápida de las propiedades que tienen los objetos de un formulario, las mismas que pueden ser copiadas y pegadas.

Las características de cada elemento de un formulario son definidas por sus diversas propiedades. Existe una paleta de propiedades sensible al contexto para cada componente, basada en el tipo de objeto que es definido. Las propiedades están agrupadas lógicamente dentro de la paleta de propiedades y cada tipo de objeto tiene una hoja de propiedades diferente. La barra de desplazamiento en el lado derecho de la paleta de propiedades permite navegar por la paleta de propiedades y los botones en forma de iconos de la parte superior controlan las actividades de la paleta. Los dos primeros botones copian y pegan todas las propiedades entre la paleta de propiedades y el portapapeles. Los dos botones siguientes agregan y eliminan propiedades personalizadas de la paleta de propiedades. El botón siguiente crea una nueva clase de propiedad basada en las propiedades del objeto. El botón Inherit (Heredar) hace que la propiedad seleccionada en la actualidad herede desde la clase de propiedad del objeto. Si no se define ninguna clase para el objeto, el botón Inherit hace que la propiedad sea restaurada al valor por defecto. El siguiente botón se usa cuando se selecciona un objeto desde el Navegador de Objetos. El botón cambia entre mostrar la unión de todas las propiedades asociados con todos los objetos seleccionados o sólo las propiedades asociadas con todos los objetos (intersección). Cuando se seleccionan varios objetos en el Navegador de Objetos, un sencillo cambio a una propiedad en la paleta de propiedades es propagado a todos los objetos. Se usa el último botón para congelar (freeze) o descongelar (unfreeze) el modo de sincronización de la hoja de propiedades. Cuando la ventana está descongelada (por defecto), siempre que se selecciona un objeto nuevo, la hoja de propiedades está sincronizada para mostrar el nuevo objeto. Sin embargo, si está congelada no se produce la sincronización hasta que se descongele.

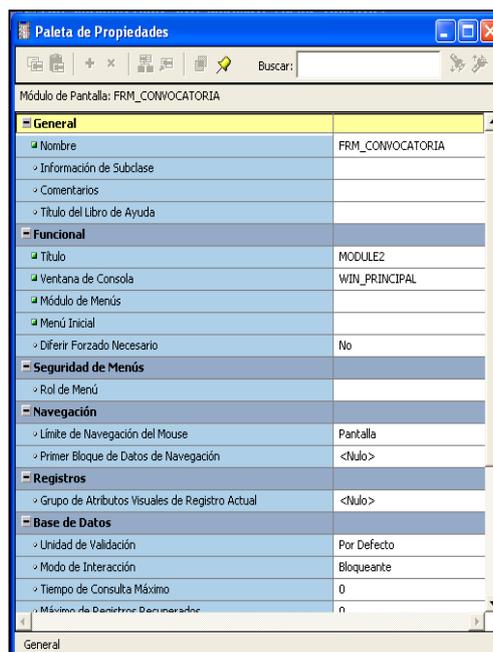


Figura 1.5. Paleta de Propiedades.
Fuente: [Propia].

- **Editor de Diseño**

Presenta el lienzo sobre el que se disponen los objetos del formulario. Esta herramienta es manejada por medio del uso de botones en forma de iconos y de opciones de menús.



Figura 1.6. Editor de Diseño.

Fuente: [Propia].

Botón	Uso
Seleccionar	Selecciona objetos en el lienzo
Rotar	Gira el elemento seleccionado
Ampliar	Hace zoom acercando o alejando el lienzo
Rectángulo	Dibuja un rectángulo en el lienzo
Línea	Dibuja una línea
Puntos suspensivos	Pone puntos suspensivos
Arco	Dibuja un arco
Polígono	Dibuja un objeto de varios lados
Polilínea	Dibuja una polilínea
Rectángulo redondeado	Dibuja un rectángulo con las esquinas redondeas
Mano alzada	Dibuja en el modo de mano alzada
Texto	Agrega texto fijo al lienzo
Marco	Agrega un marco
Botón	Crea un elemento de botón
Casilla de control	Crea un elemento de casilla de verificación
Botón de radio	Crea un elemento de botón de radio
Elemento de texto	Crea un campo de elemento de texto
Elemento de imagen	Vincula un elemento de imagen al lienzo
Elemento de gráfico	Vincula un elemento de gráfico al lienzo
Elemento mostrado	Crea un elemento de visualización
Elemento de lista	Crea un elemento de lista desplegable
Árbol Jerárquico	Crea un elemento de árbol
Lienzo con separadores	Crea un contenedor con separadores
Lienzo apilado	Crea un lienzo apilado

Tabla 1.1. Elementos del Editor de Diseño.

Fuente: [Propia].

1.1.2 Reports Developer

“El componente Reports de Developer es la parte del entorno de desarrollo con la que se realizan los informes”^[D1] a partir de tablas de la base de datos.

Los formatos en los que pueden generar los reportes son: HTML, RTF, PDF, XML, Microsoft Excel y RDF.

En este entorno se puede hacer referencia a elementos de consultas externas, y se pueden configurar y almacenar elementos de depuración.

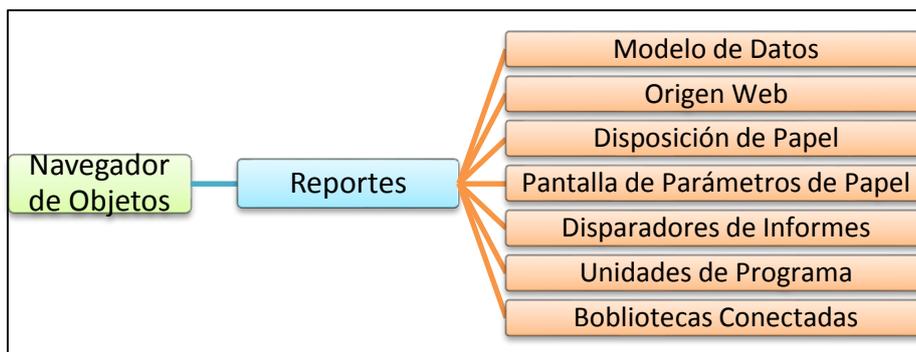


Figura 1.7. Estructura Navegador de Objetos de Reports.
Fuente: [Propia].

➤ Modelo de Datos

Es la estructura de datos y sus diferentes representaciones en el informe. El modelo de datos se crea en un editor gráfico especial.

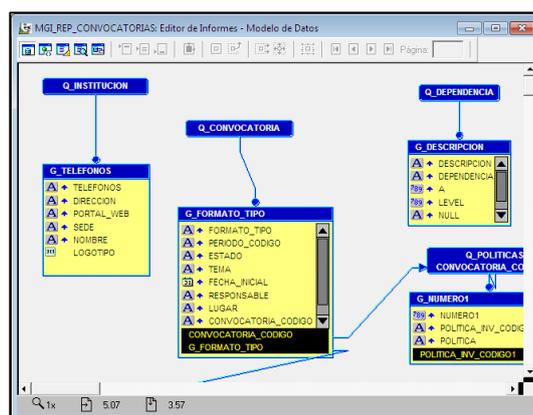


Figura 1.8. Modelo de Datos.
Fuente: [Propia].

[D1]Genero, M. (s.f.). Introducción a Oracle Developer, Recuperado el 28 Octubre del 2012, de <http://es.scribd.com/doc/101126993/Manual-Developer>

En el modelo de datos se definen **parámetros** a los que se puede hacer referencia desde el código PL/SQL y cualquier otra cosa que acepte valores de datos como entrada.

Los **parámetros del sistema** son los parámetros que Reports define automáticamente y los **parámetros de usuario** son los que define uno mismo.

Una **consulta** es una sentencia SQL que devuelve los valores, la consulta se puede incorporar en un informe o se puede utilizar un elemento de consulta externa; éste es simplemente texto SQL en un archivo independiente que se puede compartir entre aplicaciones.

El **grupo** identifica los registros que devuelven las consultas en el informe como un grupo repetido de registros, existiendo una jerarquía de grupos.

Las **columnas de base de datos** son las columnas de la lista SELECT de la consulta, las **columnas de fórmulas** son columnas que se procesan utilizando bloques PL/SQL, las **columnas de resumen** acumulan información resumida de múltiples registros en el formulario y las **columnas de almacenamiento** son columnas que se definen para ser rellenadas con un disparador, una fórmula, en vez de a partir de datos o resúmenes estándar.

Los **enlaces de datos** son enlaces utilizados en informe maestro detalle para enlazar un grupo de filas a otro grupo de filas.

➤ Disposición de Papel

La disposición de papel es la estructura gráfica del informe. Cada informe tiene una cabecera, un pie de formulario y un cuerpo intercalado entre ellos. Existen diversas variedades de composiciones de informes: tabular, maestro-detalle, formulario, carta, etiqueta de correo, etc.

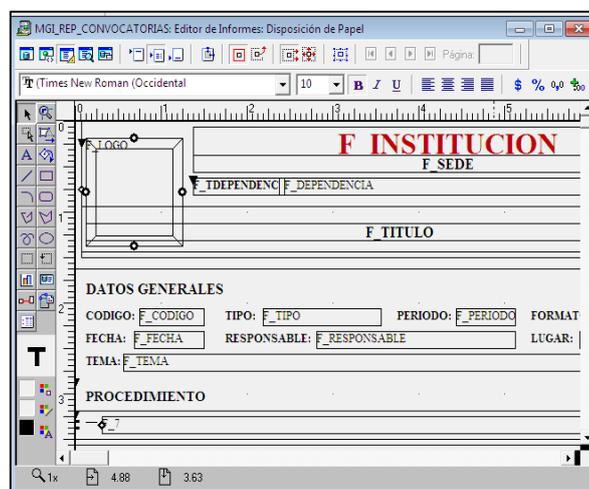


Figura 1.9. Disposición de Papel.
Fuente: [Propia].

➤ Pantalla de Parámetros de Papel

Para la asignación de valores a los parámetros en tiempo de ejecución se tiene un formulario por defecto o se puede crear uno completamente diferente como un formulario de parámetros.

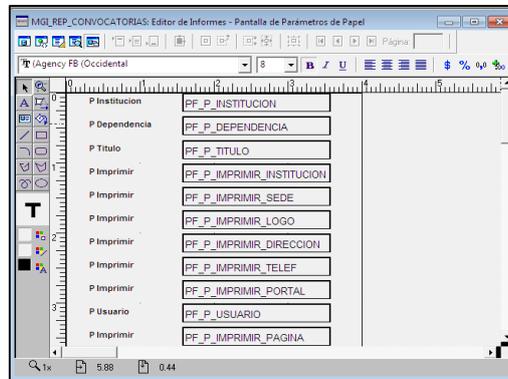


Figura 1.10. Pantalla de Parámetros de Papel.
Fuente: [Propia].

➤ Disparadores de Informes

Son bloques de código PL/SQL que se ejecutan antes o después del informe.

➤ Unidades de Programa

Son bloques de código PL/SQL llamados funciones definidos por defecto en Reports Builder.

➤ Bibliotecas Conectadas

Son enlaces a librerías o bibliotecas .pll donde se encuentren funciones, procedimientos o paquetes.

1.1.3 Designer

“Oracle Designer es una herramienta para el diseño de un sistema de información y la generación de la misma. Después de generar un sistema de información es capaz de modificar el código generado con Oracle Developer Suite”^[W2]

“Es una herramienta que permite analizar y diseñar los requerimientos de negocios para sistemas cliente/servidor que satisfagan sus requerimientos”^[D3].

^[W2]WIKIPEDIA. (s.f.). Oracle Designer. Recuperado el 25 de 11 del 2011, de http://en.wikipedia.org/wiki/Oracle_Designer

^[D3]DASGBD. (2007). Introducción a Designer. Recuperado el 28 de Octubre del 2012, de [http://alarcos.inf-cr.uclm.es/doc/aplicabdd/Documentos/teoria/INTRODUCCIÓN A DESIGNER.pdf](http://alarcos.inf-cr.uclm.es/doc/aplicabdd/Documentos/teoria/INTRODUCCIÓN_A_DESIGNER.pdf)

➤ **Facilidades**

Incorpora soporte para el proceso de modelado de negocio, para el análisis de sistemas, para el diseño de software y para la generación de sistemas.

Abarca el ciclo de vida del desarrollo de sistemas completo, generando formularios, informes y menús totalmente operativos y libres de error, además del código SQL necesario para crear los objetos de la base de datos.

Posee un repositorio multiusuario y está fuertemente relacionado con Oracle Developer.

Permite a las organizaciones diseñar rápidamente desarrollar sistemas cliente/servidor adaptables a las necesidades cambiantes de negocio^[W3].

➤ **Componentes**

Oracle Designer está compuesto por una serie de componentes donde “la jerarquía de herramientas nos permite ir de un sistema global a bases de datos reales y programas”^[D4].

➤ **Modelador de Requerimientos del Sistema**

- ✓ **Modelador de Procesos (PM, Process Modeller):** Permite realizar el diagrama de flujos de datos existentes entre procesos y almacenes de datos.
- ✓ **Diagramador de Jerarquías de Funciones (FHD, Function Hierarchy Diagrammer):** Muestra los diferentes niveles de procesos o funciones del sistema en un único diagrama.
- ✓ **Diagramador Entidad Relación (ERD, Entity Relationship Diagrammer):** Muestra las entidades con sus atributos y relaciones, representando los modelos lógicos de los datos.
- ✓ **Diagramador de Flujos de Datos (DFD, Data Flow Diagrammer):** Permite mostrar funciones, flujos de datos y almacenes. Tiene asociación de los elementos de los datos (entidades y atributos) con funciones, flujos y almacenes.

^[W3]ECURED. (2002). Oracle Designer. Recuperado el 14 de Diciembre del 2012, de http://www.ecured.cu/index.php/Oracle_Designer

^[D4]DASGBD. (2006-2007). Componentes de Designer, Recuperado el 14 de Diciembre del 2012, de <http://alarcos.inf-cr.uclm.es/doc/aplicabdd/Documentos/teoria/Tema%20-Componetntes%20de%20Designer.pdf>

➤ **Generación del Diseño Preliminar**

- ✓ **El Transformador del Diseño de BD:** Realiza la transformación del diseño conceptual de la BD al diseño lógico, trata relaciones N: M.
- ✓ **El Transformador de Diseño de la Aplicación:** Busca los procesos diseñados en la jerarquía de procesos mirando cual es el uso que hace de las entidades del sistema. Aplica reglas bien documentadas: Si no se hace ningún uso de ninguna entidad, se considera que es un proceso manual. Si solo hace operaciones de lectura, se considera que es un informe. Todos los módulos transformados por esta herramienta se consideran candidatos. El diseñador, en posteriores pasos, decidirá cuales son de utilidad y cuáles no.

➤ **Diseño y Generación del Sistema**

En esta fase podemos utilizar los objetos diseñados en módulos anteriores, o crearlos directamente aquí. Esta segunda aproximación es útil para crear prototipos rápidos de formularios e informes.

La incorporación en este módulo de ambas fases es consecuente con la política de Designer para generar código: Se diseña el módulo, se genera, se evalúa, se determinan mejoras, se vuelve a rediseñar y se vuelve a generar, hasta alcanzar la solución óptima.

El editor de diseño se divide en cuatro secciones:

- ✓ **Server Model:** Es donde realizamos el diseño y generación de la base de datos del sistema. La manipulación de datos podemos realizarla directamente sobre la jerarquía de objetos o utilizando el diagramador del server model (*Server Model Diagram*).
- ✓ **Aplicaciones de Módulo:** Se centra en el diseño y generación de aplicaciones. Tiene una estructura muy similar al modelo del servidor. El código PL/SQL escrito aquí se implementará en el cliente. Designer está diseñado para la creación de sistemas orientados a la arquitectura cliente/servidor.

➤ **Utilidades de Repositorio**

Estas utilidades asisten a lo largo de todo el ciclo de vida del desarrollo de sistemas para comprobar, cambiar o complementar el trabajo de repositorio que se efectúa con los diagramadores, generadores y otras utilidades de Designer.

- ✓ **Navegador de Objetos del Repositorio (RON):** Permite acceder y cambiar las propiedades de cualquier objeto del repositorio.

- ✓ **Diagramador de Matrices (MXD):** Presenta la información en formato de tablas con filas, columnas y valores en las intersecciones. Los valores de las intersecciones se pueden examinar y cambiar, guardando los cambios en el repositorio.
- ✓ **Informes de Repositorio:** Proporciona la posibilidad de visualizar los datos del repositorio como un informe. Se proporcionan más de 100 informes predefinidos, organizados y presentados con parámetros variables que se introducen para definir la salida.
- ✓ **Retroencaje de Tabla a Entidad:** La utilidad retroencaje de tabla a entidad encuentra las tablas existentes que no tengan una entidad base y crea entidades y atributos para ellas.
- ✓ **Utilidades de Ingeniería Inversa.:** Se puede efectuar una ingeniería inversa de objetos de la base de datos y de aplicaciones existentes para trasladarlas al repositorio.
- ✓ **Utilidades de Reconciliación.:** La Utilidad de reconciliación compara los objetos definidos en el repositorio con lo reflejado en los front-end de Designer. Los front-end solo reflejan una imagen del repositorio, y esa imagen puede ser incorrecta si otra herramienta ha accedido al repositorio y ha cambiado la definición de algunos objetos.
- ✓ **Interfaz Programativa de Aplicaciones (API):** Es un método documentado para insertar y modificar datos del repositorio.
- ✓ **Utilidades del Sistema de Aplicaciones.:** Existen otras actividades de las utilidades de repositorio, específicas de los sistemas de aplicaciones, que pueden ser desarrolladas mediante el Navegador de objetos del repositorio.
 - Compartición de objetos entre sistemas de aplicaciones.
 - Copia de objetos desde un sistema de aplicaciones a otro.
 - Transferencia de la propiedad de objetos desde un sistema de aplicaciones a otro.
 - Concesión de derechos sobre el sistema de aplicaciones a otro usuario del repositorio.
 - Archivo y restauración de sistemas de aplicaciones.
 - Renombrado, eliminación, cambio de versión y transferencia de propiedad del sistema de aplicaciones mismo.
 - Carga, descarga, comprobación de entrada y salida de objetos para trabajar con ellos en otros sistemas.
 - Desarrollo de otras acciones sobre los objetos.

- ✓ **Utilidad de Administración del Repositorio:** La utilidad de administración del repositorio se utiliza para gestionar el repositorio mismo, con todos sus sistemas de aplicaciones. Esta utilidad se puede usar para instalar, actualizar y realizar copias de seguridad de una instancia del repositorio, y para conceder y quitar permisos de acceso a usuarios. Además, se pueden añadir propiedades a elementos existentes, o incluso añadir tipos de elementos al repositorio, mediante una facilidad llamada extensibilidad de usuario.



Figura 1.11. Pantalla Componentes de Oracle 10g Designer.
Fuente: [Propia].

1.2 LENGUAJE PL/SQL

El desarrollo de una aplicación de BDD requiere de la utilización de estructuras alternativas, repetitivas, de declaración de datos similares a las que se pueden encontrar en lenguajes procedurales tales como C, C++, o Pascal. Esta construcción es necesaria para implementar tipos complejos de datos y algoritmos. El principal problema con el que se encuentra el lenguaje declarativo SQL es la ausencia de dicho tipo de estructuras y para superar este inconveniente fue creado PL/SQL (Procedural Language/SQL).

PL/SQL es un lenguaje estructurado en bloques. Cada bloque es una unidad de programa que puede ser o no etiquetada mediante una cabecera de bloque, siendo posible la existencia de bloques anidados. Los bloques que crean un procedimiento, una función, o un paquete deben ser etiquetados. Un bloque PL/SQL consta de dos secciones: una sección declarativa opcional y una sección que contiene sentencias PL/SQL. Dentro de esta última podemos incluir una parte opcional para el manejo de excepciones.

1.2.1 Características y Ventajas

- Es una extensión del lenguaje SQL realizada por Oracle.

- Es un avanzado lenguaje de programación de cuarta generación (4GL).
- Tiene una alta integración con la base de datos Oracle. y Oracle Developer Suite.
- Provee de varias características de la ingeniería de software: encapsulación, sobrecarga, tipos de datos complejos, excepciones, reutilización de código.
- Alta integración con SQL:
 - Permite la manipular datos, cursores, transacciones, funciones y operadores SQL,
 - Soporta todos los tipos de datos SQL, por lo que no es necesario ningún tipo de conversión entre las aplicaciones y la base de datos.
- Ejecución de SQL estático y dinámico.
- Programas de utilidad para administración y control de la información.
 - Se pueden desarrollar programas completos que permitan realizar tareas administrativas y de control.
 - Algunas características de las bases de datos como triggers, y procedimientos almacenados, están escritos en PL/SQL.
- Acceso a paquetes pre-definidos. Oracle incluye varios paquetes utilitarios que pueden invocarse desde las rutinas PL/SQL:
 - DBMS_ALERTS usado en triggers.
 - DBMS_FILE para lectura y escritura de archivos del sistema operativo.
 - DBMS_HTTP para realizar llamadas al protocolo HTTP.
 - DBMS_OUTPUT para desplegar información desde bloques PL/SQL y subprogramas.
 - DBMS_PIPE para comunicarse sobre tuberías nombradas.
- Soporte a programación orientada a objetos.
- Soporte para el desarrollo de aplicaciones y páginas web.

1.2.2 Herramientas y Clientes

- Toad para Oracle.
- Oracle SQL Developer.
- Oracle JDeveloper 10g
- Oracle iSQL*Plus.

- SQL Plus.
- Instant Client.

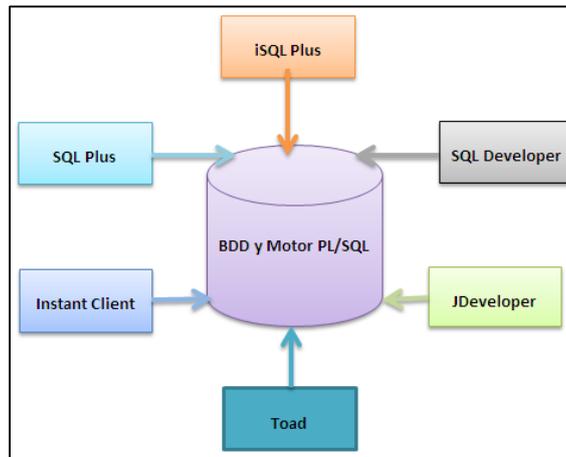


Figura 1.12. Herramientas y Clientes.
Fuente: [Propia].

1.2.3 Sintaxis Bloque PL/SQL

La sintaxis de un bloque PL/SQL es la siguiente:

```
<cabecera_bloque>  
[DECLARE  
    <Constantes>  
    <Variables>  
    <Cursores>  
    <Excepciones definidas por el usuario>]  
BEGIN  
    <Sentencias PL/SQL>  
    [EXCEPTION  
    <Manejo de excepciones>]  
END;  
<cabecera_bloque>
```

La cabecera del bloque especifica si el bloque PL/SQL es un procedimiento, una función o un paquete. Si no se especifica cabecera, se dice que el bloque es anónimo (anonymous block). En los bloques anidados el ámbito de las variables declaradas es análogo al ámbito de las variables en lenguajes de programación tales como C o Pascal.

1.2.4 Declaraciones

Las constantes, variables, cursores y excepciones que son usadas en un bloque PL/SQL deben ser declarados en su sección declarativa.

➤ Variables y Constantes

Las variables y constantes deben ser declaradas de la siguiente manera:

```
<nombre_variable> [CONSTANT] <tipo_dato> [NOTNULL] [:=<expresión>];
```

Los tipos de datos validos son los mismos que en SQL. Los datos booleanos pueden ser solo verdadero, falso, o nulo. La cláusula *NOT NULL* requiere que la variable declarada tenga un valor diferente de nulo. La *<expresión>* es utilizada para inicializar una variable.

Si no hay ninguna expresión especificada el valor null es asignado a la variable. La cláusula *[CONSTANT]* hace que una vez que el valor ha sido asignado a la variable, éste no puede ser modificado. Ejemplo:

```
DECLARE
    FECHA_INCIO           DATE;
    TITULO_COMPLETO      VARCHAR2 (1000):='SISTEMA DE GESTION';
    ASIGNACION_CERRADA   BOOLEAN;
    MONTO_PROYECTO      CONSTANT NUMBER (3,2)=1,5;
BEGIN
...
END;
```

➤ Declaración Anclada

Permite declarar el tipo de una variable asignándole a ésta, el tipo de dato asignado previamente a una columna de la tabla. Ejemplo:

```
MGI_TAB_PROYECTOS.CODIGO %TYPE;
```

Aquí se refiere al tipo de dato que tiene la columna *CODIGO* en la tabla *MGI_TAB_PROYECTOS*.

➤ Conjuntos de Atributos

Podemos declarar conjuntos de atributos que puedan almacenar una fila completa de una tabla dada. Ejemplo:

```
MGI_TAB_CONVOCATORIA %ROWTYPE;
```

Especifica un registro apropiado para almacenar todos los valores de los atributos de una fila completa de la tabla *MGI_TAB_CONVOCATORIA*. Normalmente, los registros así definidos se utilizan en combinación con un cursor. Se puede acceder a un campo en un registro utilizando *<nombre registro>.<nombre columna>*. De este modo realizamos la siguiente declaración:

```
DECLARE
CADENA MGI_TAB_CONVOCATORIA %ROWTYPE;
```

El tema de la convocatoria se referencia como *CADENA.TEMA*.

➤ **Cursores**

Una declaración de un cursor especifica un conjunto de filas resultado de una consulta, que pueden ser procesadas secuencialmente fila a fila utilizando la sentencia *FETCH*. Una declaración de un cursor tiene la forma:

```
CURSOR <nombre_cursor> [(<lista_parametros>)] IS <sentencia SELECT>;
```

El nombre del cursor es un identificador no declarado, no pudiendo ser el nombre de ninguna variable PL/SQL.

➤ **Parámetros**

Un parámetro tiene la forma:

```
<nombre_parametro> <tipo_parametro>
```

Los tipos de parámetros posibles son *CHAR*, *VARCHAR*, *NUMBER*, *DATE* y *BOOLEAN* así como sus subtipos correspondientes tal como *INTEGER*. Los parámetros son utilizados para asignar valores a las variables que utilizan las sentencias *SELECT*. Ejemplo:

```
CURSOR ARBOL_LOCALIDADES (LOCALIDAD VARCHAR2, TIPO VARCHAR2) IS  
SELECT CODIGO, DESCRIPCION  
FROM INS_TAB_LOCALIDADES L  
WHERE L.LOCALIDAD_CODIGO=LOCALIDAD  
AND EXITSTS (SELECT * FROM INS_TAB_LOCALIDADES L  
WHERE L.FUNCION='G'AND L.TLOCALIDAD_CODIGO=TIPO);
```

Si alguna de las filas seleccionadas fuera modificada en el bloque PL/SQL, la cláusula *FOR UPDATE* [(*<columna(s)>*)] debería ser añadida al final de la declaración del cursor. En este caso, las filas seleccionadas son bloqueadas y no pueden ser accedidas por otros usuarios hasta que sea ejecutado un comando *COMMIT*. Antes de utilizar un cursor declarado previamente, éste debe estar abierto. Una vez procesadas las filas definidas en él, debe de ser cerrado.

➤ **Excepciones**

Las excepciones son utilizadas para procesar de forma controlada los errores y advertencias que ocurren durante la ejecución de sentencias PL/SQL. Algunas excepciones, como *ZERO_DIVIDE*, son definidas internamente. Otras excepciones pueden ser especificadas por el usuario al final de un bloque PL/SQL. Para que el usuario defina las excepciones necesita declararlas utilizando:

<nombre_excepcion> EXCEPTION

1.2.5 Elementos del Lenguaje

PL/SQL presenta varios elementos como la asignación de valores a variables, estructuras repetitivas, estructuras alternativas, llamadas a procedimientos y funciones, etc. Sin embargo, PL/SQL no permite comandos del lenguaje de definición de datos DDL (Data Definition Language) tales como la creación de tablas (*CREATE TABLE*).

➤ Asignación de Valores a Variables

En PL/SQL hay varias alternativas para la asignación de un valor a una variable. El camino más sencillo para asignar un valor a una variable es:

```
DECLARE
    CONTADOR INTEGER:=0;
BEGIN
    CONTADOR:=CONTADOR +1;
END;
```

Los valores asignados a una variable pueden ser recuperados de la base de datos usando la sentencia *SELECT*:

```
SELECT <columna(s)> INTO <lista_variables> FROM <tabla(s)> WHERE <condición>;
```

La sentencia *SELECT* recupera a lo sumo una fila. Si no ocurre esto, no es posible asignar los valores del atributo a la lista de variables especificada y se produciría un error de ejecución. Si el *SELECT* recupera más de una fila, en lugar de éste, deberá ser utilizado un cursor. Además, los tipos de datos de las variables especificadas deben concordar con los valores de atributo recuperados. Para la mayoría de los tipos de datos, PL/SQL realiza una conversión automática.

Tras la palabra reservada *INTO*, en lugar de una lista de variables podemos incluir un registro. Este caso, la sentencia *SELECT* debe recuperar como máximo una única fila:

```
DECLARE
    PROYECTO MGI_TAB_PROYECTOS %ROWTYPE;
    MONTO MGI_TAB_PROYECTOS.MONTO_PROYECTO %TYPE;
BEGIN
    SELECT CODIGO, TITULO_COMPLETO, DURACION INTO PROYECTO
    FROM MGI_TAB_PROYECTOS WHERE CODIGO='0000000001';
    SELECT MAX (MONTO_PROYECTO) INTO MONTO
    FROM MGI_TAB_PROYECTOS;
END;
```

➤ Estructuras de Control

Para emplear la estructura **IF** usamos la siguiente sintaxis:

```
IF <condición> THEN
    <secuencia_sentencias>;
ENDIF;
```

A continuación se muestra un ejemplo:

```
DECLARE
    ESTADO VARCHAR2 (10):='01';
BEGIN
    IF (ESTADO='01') THEN
        UPDATE MGI_TAB_PROYECTOS PROY
        SET ESTADO_PRO_CODIGO='08'
        WHERE PROY.CODIGO=PROYECTO;
    ELSE
        UPDATE MGI_TAB_PROYECTOS PROY
        SET ESTADO_PRO_CODIGO='09'
        WHERE PROY.CODIGO=PROYECTO;
    END IF;
    COMMIT;
END;
```

Uso de la estructura de control **CASE**, sintaxis:

```
CASE <condición>
WHEN <condición> THEN <sentencia>
...
ELSE <sentencia>
END;
```

Ejemplo:

```
DECLARE
    GRADE CHAR (1):='B';
    APPRAISAL VARCHAR2 (20);
BEGIN
    APPRAISAL :=
    CASE GRADE
        WHEN 'A' THEN 'EXCELLENT'
        WHEN 'B' THEN 'VERY GOOD'
        WHEN 'C' THEN 'GOOD'
        WHEN 'D' THEN 'FAIR'
        ELSE 'NO' SUCH GRADE'
    END;
    DBMS_OUTPUT.PUT_LINE ('GRADE ' || GRADE || ' IS ' || APPRAISAL);
END;
```

La sintaxis de un bucle **WHILE** es la siguiente:

```
WHILE <condición> LOOP
    <secuencia_sentencias>;
```

END LOOP;

En general, el etiquetado del bucle es opcional, aunque cuando se presentan varios bucles continuos (incondicionales) anidados, resulta obligatorio el etiquetado, ya que es necesario incluir la sentencia:

EXIT <nombre_etiqueta>;

Por tanto, el número de iteraciones a través de un bucle *WHILE* no es conocido hasta que el bucle finaliza. Ejemplo:

```
DECLARE
    SAL INTEGER;
BEGIN
    SAL :=5000;
    WHILE SAL <=15000 LOOP
        SAL := SAL +5000;
    END LOOP;
END;
```

La sintaxis de un bucle **FOR** es la siguiente:

```
FOR <índice> IN [REVERSE] <límite_inferior>...<límite_superior> LOOP
    <secuencia_sentencias>;
END LOOP;
```

El contador de bucle *<índice>* esta implícitamente declarado. El ámbito del contador del bucle es sólo el bucle *FOR*. Esto anula el alcance de algunas variables que tienen el mismo nombre fuera del bucle. Dentro el bucle *FOR*, *<índice>* puede ser referenciado como una constante, *<índice>* puede aparecer en expresiones, pero no es posible asignarle valores. El uso de la palabra reservada *REVERSE*, provoca un incremento negativo del índice, desde el límite superior (*<límite_superior>*) hasta el inferior (*<límite_inferior>*).

Ejemplo:

```
DECLARE
    S INTEGER;
BEGIN
    FOR I IN 1..100 LOOP
        S :=(i * (i +1) * (2*i +1))/6;
    END LOOP;
END;
```

1.2.6 Manejo de Cursores

Antes de que un cursor pueda ser utilizado, debe ser abierto usando la sentencia *OPEN*.

OPEN <nombre_cursor> *INTO* [<lista_parámetros>];

La sentencia *SELECT* asociada es entonces procesada, y el cursor hace referencia a la primera fila seleccionada. Las filas seleccionadas pueden ser entonces procesadas una a una usando el comando *FETCH*.

FETCH <nombre_cursor> *INTO* <lista_variables>;

El comando *FETCH* asigna los valores de los atributos seleccionados de la fila actual a la lista de variables. Después de ejecutar el comando *FETCH*, el cursor avanza a la siguiente fila del conjunto seleccionado. Debemos tener en cuenta que las variables de la lista deben de tener los mismos tipos de datos que los valores seleccionados. Después de que todas las filas hayan sido procesadas, el cursor debe de ser deshabilitado mediante el comando *CLOSE*.

CLOSE <nombre_cursor>;

El siguiente ejemplo muestra como un cursor es utilizado junto a un bucle:

```
DECLARE
    CURSOR ARBOL_LOCALIDADES IS
    SELECT * FROM INS_TAB_LOCALIDADES;
    REG_LOCALIDAD INS_TAB_LOCALIDADES %ROWTYPE;
    SAL_LOCALIDAD INS_TAB_LOCALIDADES.DESCRIPCION %TYPE;
BEGIN
    OPEN ARBOL_LOCALIDADES;
    LOOP
        FETCH ARBOL_LOCALIDADES INTO REG_LOCALIDAD;
        EXIT WHEN ARBOL_LOCALIDADES %NOT FOUND;
        SAL_LOCALIDAD:=REG_LOCALIDAD.DESCRIPCION;
        <SECUENCIA DE SENTENCIAS>
    END LOOP;
    CLOSE ARBOL_LOCALIDADES;
END;
```

Cada bucle puede ser completado incondicionalmente usando la cláusula *EXIT*.

EXIT [<etiqueta_bloque>] [*WHEN* <condición>]

El uso de *EXIT* sin una etiqueta de bloque causa el fin de la ejecución del bucle que contiene la sentencia *EXIT*. La condición puede ser una simple comparación de valores. En la mayoría de los casos, la condición se refiere a un cursor. En el ejemplo anterior, %*NOTFOUND* es un operador que se evalúa como falso si el último comando *FETCH* ejecutado ha leído una fila. Si no ha sido posible leer una fila al haber sido ya todas procesadas, %*NOTFOUND* devuelve falso. El valor de <nombre_cursor> %*NOTFOUND* es nulo antes de que la primera fila sea procesada con un *FETCH*. El operador %*FOUND* es la oposición lógica de %*NOTFOUND*.

El cursor de los bucles *FOR* puede ser usado para simplificar el uso del cursor:

```
FOR <nombre_registro> IN <nombre_cursor> [(<lista_parámetros>)] LOOP
    <secuencia_sentencias>
END LOOP;
```

Con esta sintaxis, se declara de forma implícita el registro apropiado para almacenar una fila del cursor. Además, este bucle ejecuta implícitamente un *FETCH* en cada iteración, así como un *OPEN* antes de que se entre en el bucle y un *CLOSE* después de que el bucle finalice. Si se intenta ejecutar un bucle con un cursor que no ha seleccionado ninguna fila, éste finaliza automáticamente.

Esto es igualmente posible al especificar una consulta en lugar de *<nombre_cursor>* en un bucle *FOR*:

```
FOR <nombre_registro> IN (<sentencia_select>) LOOP
    <secuencia_sentencias>
END LOOP;
```

Es decir, un cursor no necesita ser especificado antes de entrar el bucle, pero es definido en la sentencia *SELECT*. Ejemplo:

```
FOR REG_SAL IN (SELECT MONTO_PROYECTO-MONTO_GASTADO SALDO
FROM MGI_TAB_PROYECTOS) LOOP
...;
END LOOP;
```

SALDO es un alias para la expresión en la sentencia *SELECT*. Así, en cada iteración sólo se procesa una única fila. El registro *REG_SAL*, que está implícitamente definido, contiene sólo una entrada a la que podemos acceder con *REG_SAL.SALDO*. Los alias, por supuesto, no son necesarios si son seleccionados atributos exclusivamente, es decir, si la sentencia *SELECT* no contiene ni operadores aritméticos ni funciones agregadas.

Para las estructuras de control alternativas, PL/SQL ofrece la estructura *IF-THEN-ELSE*

```
IF <condición> THEN <secuencia_sentencias>
[ELSIF] <condición> THEN <secuencia_sentencias>
...
[ELSE] <secuencia_sentencias>
END IF;
```

Empezando con la primera condición, si ésta es verdadera, su correspondiente secuencia de sentencias son ejecutadas, de otra forma el control pasaría a la próxima condición. Así la conducta de este tipo de sentencias PL/SQL es análoga a las sentencias *IF-THEN-ELSE* en los lenguajes de programación imperativos.

Excepto en los comandos DDL como *CREATE TABLE*, todos los tipos de sentencias SQL pueden ser utilizadas en los bloques PL/SQL, en particular *DELETE*, *INSERT*, *UPDATE*, y *COMMIT*. Nótese que en PL/SQL sólo se permite la sentencia *SELECT* del tipo *SELECT<columna(s)> INTO <lista_variables>*, es decir, los valores de atributo seleccionados sólo pueden ser asignados a variables. El uso de la sentencia *SELECT SQL* provoca un error de sintaxis.

Si las sentencias *UPDATE* o *DELETE* se utilizan en combinación con un cursor, estos comandos pueden ser restringidos a la fila en proceso. Para ello, se añade la cláusula *WHERE CURRENT OF <nombre_cursor>*, tal y como se muestra en el siguiente ejemplo.

```
DECLARE
    DOCENTE MGI_TAB_INVESTIGADORES.DOCENTE_CEDULA %TYPE;
    CURSOR INV_CUR (COD_INV VARCHAR2) IS SELECT FECHA_INCIO
    FROM MGI_TAB_INVESTIGADORES WHERE CODIGO = COD_INV
    FOR UPDATE OF FECHA_INCIO;
BEGIN
    SELECT DOCENTE_CEDULA INTO DOCENTE
    FROM MGI_TAB_INVESTIGADORES WHERE TIPO_INT_CODIGO = '003';
    FOR REG_DOC IN INV_CUR (DOCENTE) LOOP
        UPDATE MGI_TAB_INVESTIGADORES
        SET FECHA_INCIO = REG_DOC.FECHA_INCIO * 1.05
        WHERE CURRENT OF INV_CUR;
    END LOOP;
    COMMIT;
END;
```

1.2.7 Manejo de Excepciones

Un bloque PL/SQL puede contener sentencias que especifique rutinas de manejo de excepciones. Cada error o advertencia durante la ejecución del bloque PL/SQL genera una excepción.

Podemos distinguir excepciones definidas por el sistema y excepciones definidas por el usuario.

Las **excepciones definidas por el sistema** son automáticamente generadas cuando se produce un error o advertencia. Por el contrario, las **excepciones definidas por el usuario**, deben ser generadas implícitamente mediante una secuencia de sentencias utilizando la cláusula:

```
RAISE <nombre_excepción>
```

Las rutinas del manejo de excepciones son implementadas por el usuario al final del bloque y tras la palabra reservada *EXCEPTION*:

```
WHEN <nombre_excepción> THEN <secuencia_sentencias>;
```

Los errores más comunes que pueden darse durante la ejecución de los programas PL/SQL son manejados por excepciones definidas por sistema. La siguiente tabla muestra algunas de ellas:

Nombre	Número	Observaciones
CURSOR_ALREADY_OPEN	ORA-06511	Se ha intentado abrir un cursor que está ya abierto.
INVALID_CURSOR	ORA-01001	Se ha intentado hacer un fetch sobre un cursor cerrado.
NOT_DATA_FOUND	ORA-01403	Una sentencia select ... into o fetch no devuelve ninguna fila.
TOO_MANY_ROWS	ORA-01422	Una sentencia select ... into o fetch no devuelve más de una fila.
ZERO_DIVIDE	ORA-01476	Se ha intentado dividir un número entre 0.

Tabla 1.2. Excepciones del Sistema.

Fuente: [Propia].

A continuación se presenta un ejemplo:

DECLARE

HOR_DED MGI_TAB_INVESTIGADORES.HORAS_DEDICACION %TYPE;
CED_DOC MGI_TAB_INVESTIGADORES.DOCENTE_CEDULA %TYPE
HORAS_DEMASIADO_ALTO EXCEPTION;

BEGIN

SELECT DOCENTE_CEDULA, HORAS_DEDICACION INTO CED_DOC, HOR_DED
FROM MGI_TAB_INVESTIGADORES WHERE DOCENTE_CEDULA = '1708465669';
IF HOR_DED > 40 THEN
 RAISE HORAS_DEMASIADO_ALTO
ELSE
 UPDATE MGI_TAB_INVESTIGADORES SET HORAS_DEDICACION=30;
END IF;
EXCEPTION
 WHEN NO_DATA_FOUND THEN
 ROLLBACK;
 WHEN HORAS_DEDICACION THEN
 COMMIT;

END;

1.2.8 Procedimientos

PL/SQL proporciona un lenguaje sofisticado para construir procedimientos de programa que pueden ser invocados por otros bloques PL/SQL y por otros procedimientos y funciones.

La sintaxis para la definición de procedimientos es:

CREATE [OR REPLACE] PROCEDURE <nombre_procedimiento>
(<lista_parámetros>)] IS <declaración>

```
BEGIN
    <secuencia_sentencias>
    [EXCEPTION <rutinas_manejo_excepciones>]
END [<nombre_procedimiento>];
```

La cláusula opcional *OR REPLACE* vuelve a crear el procedimiento. Ejemplo:

```
CREATE OR REPLACE PROCEDURE
UTNDB.MGI_PRO_CAMBIAR_FECHA_EN
(TABLA VARCHAR2, CAMPO1 VARCHAR2, CRITERIO1 VARCHAR2,
CAMPO2 VARCHAR2, CRITERIO2 VARCHAR2)
IS CADENA VARCHAR2 (600);
BEGIN
    CADENA:='UPDATE '||TABLA||' SET '||TABLA||'.'||CAMPO1||' = '||''''||CRITERIO1||''''||'
WHERE '||TABLA||'.'||CAMPO2||' = '||''''||CRITERIO2||'''';
    EXECUTE IMMEDIATE CADENA;
    COMMIT;
    EXCEPTION
    WHEN NO_DATA_FOUND THEN
        NULL;
    WHEN OTHERS THEN
        RAISE;
END MGI_PRO_CAMBIAR_FECHA_EN;
```

1.2.9 Funciones

Una función puede ser especificada de la siguiente manera:

```
CREATE [OR REPLACE] FUNCTION <nombre_función>
[(<lista_parámetros>)]
BEGIN
    <secuencia_sentencias>
    [EXCEPTION <rutinas_manejo_excepciones>]
END [<nombre_función>];
```

La cláusula opcional *OR REPLACE* vuelve a crear la función. Ejemplo:

```
CREATE OR REPLACE FUNCTION
UTNDB.MGI_FUN_ULTIMO_PERIODO
(PROYECTO VARCHAR, PERIODO VARCHAR)
RETURN DATE
IS ULTIMO_PERIODO DATE;
BEGIN
    SELECT MGI_TAB_PERIODO_PROYECTO.HASTA
    INTO ULTIMO_PERIODO
    FROM MGI_TAB_PERIODO_PROYECTO
    WHERE MGI_TAB_PERIODO_PROYECTO.PROYECTO_CODIGO=PROYECTO
    AND MGI_TAB_PERIODO_PROYECTO.PERIODO_CODIGO=PERIODO;
    RETURN ULTIMO_PERIODO;
END MGI_FUN_ULTIMO_PERIODO;
```

1.2.10 Borrar Procedimientos y Funciones

Un procedimiento o función puede ser borrado respectivamente con los comandos:

```
DROP PROCEDURE <nombre_procedimiento>  
DROP FUNCTION <nombre_función>
```

Ejemplo:

```
DROP PROCEDURE UTNDB.MGI_PRO_CAMBIAR_FECHA_EN;  
DROP FUNCTION UTNDB.MGI_FUN_ULTIMO_PERIODO;
```

1.2.11 Parámetros Válidos

Los parámetros válidos incluyen todos los tipos de datos. Sin embargo, para los tipos *CHAR*, *VARCHAR2* y *NUMBER* no deben de especificar longitud ni escala. Por ejemplo, el parámetro *NUMBER (6)* da un error de compilación y debe ser remplazado por *NUMBER*. Los tipos implícitos del tipo *%TYPE* y *%ROWTYPE* pueden ser utilizados sin ninguna restricción.

Los parámetros se especifican de la siguiente forma:

```
<nombre_parámetro> [IN | OUT | INOUT]  
<tipo_dato> [{:= |DEFAULT} <expresión>]
```

Las cláusulas *IN*, *OUT* e *IN OUT* especifican el modo en que es utilizado el parámetro. Por defecto, los parámetros son de modo *IN*. *IN* significa que el parámetro puede estar referenciado dentro del cuerpo del procedimiento, pero no puede ser cambiado. *OUT* significa que al parámetro se le puede asignar un valor, pero el valor del parámetro no puede ser referenciado. *IN OUT* permite ambas cosas, asignar valores a los parámetros y referenciarlos. Normalmente se usa el modo (*IN*).

1.2.12 Disparadores (triggers)

Los disparadores proporcionan una técnica procedural para especificar y mantener las restricciones de integridad. Los disparadores incluso permiten a los usuarios especificar las condiciones de integridad más complejas, ya que un disparador es esencialmente un procedimiento PL/SQL asociado con una tabla, que es automáticamente llamado por el SGBD cuando ocurre alguna modificación en la tabla. Las modificaciones de la tabla pueden incluir operaciones *INSERT*, *UPDATE*, y *DELETE*.

➤ Estructura de los Disparadores

La definición de un disparador consta de los siguientes componentes (opcionales):

- Nombre del disparador.

CREATE [OR REPLACE] TRIGGER <nombre_disparador>

- Tiempo o punto del disparador.

BEFORE/AFTER

- Eventos disparadores.

INSERT OR UPDATE [OF <columna(s)>] OR DELETE ON <tabla>

- Tipo de disparador (opcional).

FOR EACH ROW

- Restricciones disparador (solo para disparador *FOR EACH ROW*).

WHEN (<condición>)

- Cuerpo del disparador.

<Bloque PL/SQL>

La cláusula *REPLACE* vuelve a crear una definición de disparador con el mismo nombre del disparador. El nombre de un disparador puede ser elegido arbitrariamente, pero un buen estilo de programación es usar un nombre de disparador que refleje al de la tabla y el evento. Un disparador puede ser invocado antes o después del evento disparador. Un evento simple es una sentencia *INSERT*, *UPDATE* o *DELETE*; los eventos pueden ser combinados usando la conectiva lógica *OR*. Si no se especifican columnas en un disparador *UPDATE*, el disparador es ejecutado cuando la tabla es modificada. Si el disparador debe de ejecutarse cuando son actualizadas sólo determinadas columnas, estas columnas deben de ser especificadas después del evento *UPDATE*.

Para programar disparadores de forma eficiente y correcta es esencial comprender la diferencia entre un disparador a nivel de fila y un disparador a nivel de sentencia. Un disparador a nivel de fila se define utilizando la cláusula *FOR EACH ROW*. Si no aparece esta cláusula en la definición del disparador, se asume que es un disparador a nivel de sentencia. Un disparador a nivel de fila se ejecuta una vez por cada fila (antes o después) del evento. Por el contrario, un disparador a nivel de sentencia se ejecuta una única vez (antes o después) del evento, independientemente del número de las filas afectadas por el evento.

Los disparadores a nivel de fila tienen algunas características especiales de la que carecen los disparadores a nivel de sentencias. Sólo con un disparador a nivel de fila es posible acceder a los valores de los atributos de una fila antes y después de la modificación. Para un disparador *UPDATE* podemos acceder al valor que posee el atributo antes del disparo utilizando *:OLD* y

podemos acceder al valor de atributo tras el disparo utilizando *:NEW*. Para un disparador *INSERT* sólo puede ser utilizado *:NEW* y para un disparador *DELETE* sólo puede ser utilizado *:OLD*. En estos casos *:NEW* se refiere a valores de atributos de la fila insertada, y *:OLD* se refiere al valor de atributo de la columna de la fila borrada. En un disparador a nivel de fila es posible especificar comparaciones entre un nuevo y antiguo valor de atributo en el bloque PL/SQL. En general, es aconsejable usar un disparador a nivel de fila *AFTER* si la nueva fila no es modificada en el bloque PL/SQL. Los disparadores a nivel de sentencia son usados generalmente en combinación con *AFTER*.

En una definición de disparador, la cláusula *WHEN* sólo puede ser utilizada en combinación con la opción *FOR EACH ROW*. Esta cláusula es utilizada para restringir cuándo se ejecuta el disparador. Para la especificación de la condición en la cláusula *WHEN*, se siguen las mismas normas que para la cláusula *CHECK*. Las únicas excepciones son que las funciones *SYSDATE* y *USER* pueden ser utilizadas, y que es posible referirse al viejo/nuevo valor de atributo de la fila actual. En el caso anterior, el símbolo “:” no debe ser usado (*OLD* y *NEW*).

El cuerpo del disparador consiste en un bloque PL/SQL. Todos los comandos PL/SQL pueden ser utilizados en un bloque disparador, a excepción de las sentencias *COMMIT* y *ROLLBACK*. Además, la utilización adicional de la estructura *IF* permite ejecutar ciertas partes del bloque PL/SQL dependiendo del evento disparador. Para ello, existen las construcciones: *IF INSERTING*, *IF UPDATING* [(‘<columna>’)] e *IF DELETING*, que pueden ser utilizadas.

Ejemplo:

```
CREATE OR REPLACE TRIGGER MGI_TGR_COMPROBAR_PRES_PROY
AFTER INSERT OR UPDATE OF MONTO_PROYECTO ON MGI_TAB_PROYECTOS
DECLARE
    CURSOR PROY_CUR IS
        SELECT CODIGO, MONTO_PROYECTO FROM MGI_TAB_PROYECTOS;
    DNO MGI_TAB_PROYECTOS.CODIGO %TYPE;
    TOTAL_MONTO MGI_TAB_PROYECTOS.MONTO_PROYECTO %TYPE;
    PROY_MONTO NUMBER;
BEGIN
    OPEN PROY_CUR;
    LOOP
        FETCH PROY_CUR INTO DNO, TOTAL_MONTO;
        EXIT WHEN PROY_CUR %NOT FOUND;
        SELECT SUM (MONTO_PROYECTO) INTO TOTAL_MONTO
        FROM MGI_TAB_PROYECTOS WHERE CODIGO = DNO;
        IF PROY_MONTO > TOTAL_MONTO THEN
            RAISE_APPLICATION_ERROR (-20325,
            'EL TOTAL DEL MONTO DEL PROYECTO.'||TO_CHAR (DNO||
            'EXCEDE DEL PRESUPUESTO');
        END IF;
    END LOOP;
```

```
END LOOP;  
CLOSE PROJ_CUR;  
END;
```

➤ Programación de Disparadores

Los disparadores a nivel de fila son el tipo de disparador más problemático, ya que incluyen varias restricciones. Para asegurar la consistencia de lectura, Oracle realiza un bloqueo de la tabla desde el comienzo de las sentencias *INSERT*, *UPDATE*, o *DELETE*. Esto quiere decir, que otros usuarios no pueden acceder a esta tabla hasta que las inserciones/modificaciones/borrados se hayan completado satisfactoriamente. En este caso, la tabla sobre la que se realiza la operación se denomina tabla mutante. La única forma de acceder a una tabla mutante en un disparador es utilizando *:OLD.<columna>* y *:NEW.<columna>* en conexión con una fila disparador.

Ejemplo de un disparador a nivel de fila erróneo:

```
CREATE OR REPLACE TRIGGER MGI_TGR_COMPROBAR_PRES_PROY  
AFTER INSERT OR UPDATE OF MONTO_PROYECTO ON MGI_TAB_PROYECTOS  
FOREACH ROW  
DECLARE  
SUM_MON NUMBER;  
BEGIN  
SELECT SUM (MONTO_PROYECTO) INTO SUM_MON FROM MGI_TAB_PROYECTOS;  
END;
```

Si una sentencia *UPDATE* de la forma

```
UPDATE MGI_TAB_PROYECTOS SET MONTO_PROYECTO = MONTO_PROYECTO * 1.1;
```

Es ejecutada en la tabla *MGI_TAB_PROYECTOS*, el disparador de arriba es ejecutado una vez por cada fila modificada. Mientras la tabla está siendo modificada por el comando *UPDATE*, no es posible acceder a todas las filas de la tabla usando el comando *SELECT*, porque este está cerrado.

En este caso obtendremos los siguientes mensajes de error:

```
ORA-04091: tabla MGI_TAB_PROYECTOS está mutando.  
ORA-06512: en la línea 4.  
ORA-04088: error durante la ejecución del disparador.
```

La forma de acceder a la tabla, y a la fila, es usar *:OLD.<columna>* y *:NEW.<columna>*.

Reglas para la definición de integridad manteniendo disparadores:

```
Para cada restricción de integridad para cada tabla chequear  
IF la restricción puede ser comprobada a nivel de fila THEN  
IF filas comprobadas modificadas dentro del disparador THEN  
Usar disparador-fila tipo BEFORE
```

ELSE

*Usar disparador-fila tipo **AFTER***

ELSE

*Usar disparador-sentencia tipo **AFTER***

Los disparadores pueden ser utilizados también para:

- Control de acceso a los usuarios y modificaciones en ciertas tablas.
- Monitorización de operaciones sobre tablas.
- La propagación automática de las modificaciones.

1.3 ORACLE 10G

1.3.1 RDBMS

Un RDBMS¹² es un Sistema Gestor de Bases de Datos Relacionales. Se trata de un software capaz de producir, manipular y gestionar bases de datos de tipo relacional.

Es un software que se antepone a los datos de una base de datos, de modo que cualquier acceso a los datos pasa por una petición al RDBMS que éste gestiona a fin de realizar la operación más conveniente sobre esa petición.

Prácticamente es un sistema operativo diseñado para el control del acceso a los datos.

Para conseguir este control, todo RDBMS posee una serie de subsistemas que se encargan de gestionar cada servicio. Algunos de estos subsistemas son:

- **Sistema de Gestión de la Memoria:** Encargado de decidir que parte de la memoria se dedica a cada tarea del RDBMS. Su función es que haya suficiente memoria para que el RDBMS funcione eficazmente y a la vez nunca dejar menos memoria de la que necesita el sistema operativo para que la máquina funcione.
- **Gestión de Entrada y Salida:** Para conseguir que los accesos a los datos sean adecuados.
- **Procesador de Lenguajes:** Para interpretar las instrucciones SQL (o de otros lenguajes válidos) que los usuarios lanzan a la base de datos.
- **Control de Procesos:** Gestiona los programas en ejecución necesarios para el funcionamiento de la base de datos.

¹²Sistema de Gestión de Base de Datos Relacional.

- **Control de la Red:** Para gestionar las conexiones a la base de datos desde la red y evitar problemas a la base de datos en caso de desconexión.
- **Control de Transacciones:** Permite gestionar las transacciones (series de operaciones que se pueden anular o llevar a cabo al final).

1.3.2 Diccionario de Datos

El diccionario de datos agrupa los metadatos¹³ de una base de datos. En este diccionario aparecen todos los objetos de la base de datos; con su nombre, función, control de acceso (seguridad) y correspondencia física en los archivos de datos.

Cada vez que llega al gestor de bases de datos una petición sobre datos de una base de datos, el RDBMS abre el diccionario de datos para comprobar los metadatos relacionados con la petición y resolver si hay permiso de uso y donde localizar físicamente los datos requeridos.

1.3.3 Conexión a un Sistema Gestor de Base de Datos

Normalmente cualquier DBMS funciona como servidor, programa que está en ejecución esperando peticiones de conexión al sistema. En cada intento de conexión el sistema verificará qué usuario intenta conectar y si tiene permiso se produce la conexión.

En la conexión el usuario puede ejecutar peticiones sobre la base de datos en el lenguaje, o lenguajes, que el DBMS sea capaz de traducir.

Esto permite centralizar la información ya que el servidor se puede encontrar alejado del usuario que intenta acceder. De modo que el usuario puede estar en un ordenador y el servidor en otro.

➤ Conexión Local

El servidor de bdd y el usuario que intenta conectar están en la misma máquina. No hace falta control de red, pero limita el uso de la bdd a la máquina en la que el servidor está instalado.

¹³Son datos que describen otros datos.



Figura 1.13. Conexión Local.
Fuente: [Propia].

➤ Cliente / Servidor

Se trata del método más común de trabajo. El servidor de bases de datos lanza un proceso en la máquina central (servidor) desde la que se gestionan las bases de datos. Este proceso está a la escucha de nuevos usuarios, cuando estos llegan se produce una conexión que permite que el servidor y el cliente se comuniquen.

La ventaja de esta implementación es que permite centralizar el sistema de datos, lo que facilita su control. Por otro lado eso permite una accesibilidad a la base de datos desde distintas máquinas.

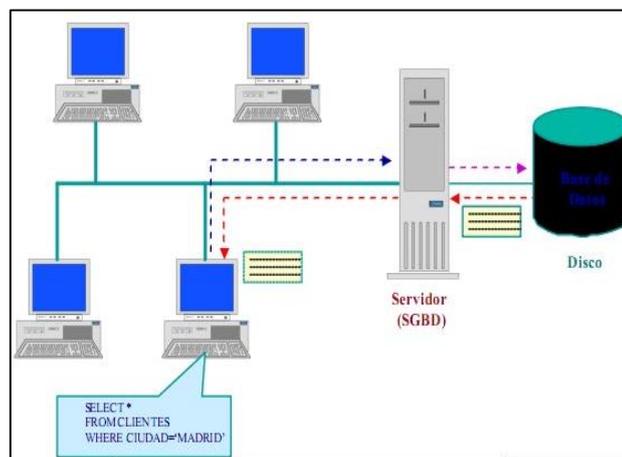


Figura 1.14. Cliente/Servidor.
Fuente: [W4].

[W4]BLOGSPOT. (s.f.). Introducción a Oracle Arquitectura Cliente/Servidor. Recuperado el 28 de Septiembre del 2012, de <http://introduccionaoracle.blogspot.com/p/arquitectura-cliente-servidor.html>

1.3.4 Herramientas de los RDBMS

Para el uso de las bases de datos, los RDBMS proporcionan diversas herramientas, que además tienen finalidades distintas en función de qué tipo de usuario las utiliza:

Herramientas de Instalación: Instaladores para facilitar la tarea de realizar la siempre difícil instalación del producto de base de datos.

Herramientas de Gestión de Red: Permiten que el gestor de base de datos sea accesible desde la red, así como gestionar el correcto flujo de información sobre la red que integra al RDBMS.

- Herramientas de conexión en el lado del cliente: Los programas que permiten a los usuarios conectar a la base de datos para lanzar las instrucciones que se deseen.
- Herramientas de desarrollo: Que facilitan la labor de crear aplicaciones para una base de datos.
- Herramientas CASE: Para poder realizar diseños completos de aplicaciones de datos.
- Herramientas de administración: Que permiten una más fácil realización de las tareas administrativas.
- Herramientas de recuperación: que permiten recuperarse en caso de desastre.
- Herramientas de copia de seguridad.
- Herramientas de importación de datos.
- Herramientas de globalización.
- Herramientas de creación de aplicaciones hacia Internet.

1.3.5 Base de Datos Objeto-Relacional

Las bases de datos relacionales incorporan sólo estructuras estáticas de datos, las llamadas tablas. Es suficiente para modelar problemas de datos muy grandes, pero no permite aplicar técnicas de programación orientada a objetos.

Hay una serie de bases de datos llamadas orientadas a objetos, donde el elemento de trabajo en lugar de la tabla es el objeto, que incorpora datos y procedimientos.

Oracle aprovecha el enfoque objeto-relacional ya que permite a los usuarios utilizar la base de datos de forma relacional, pero incorpora extensiones de las bases de datos orientadas a objetos.

Oracle soporta el enfoque orientado a objetos. El hecho de que permite los tres objetos hace que se pueda trabajar de forma relacional y añadir cualquier mejora orientada a objetos. Pero el núcleo de Oracle sigue estando pensado para el enfoque relacional.

1.3.6 Mejoras del Enfoque Orientado a Objetos

- **Tipos Abstractos:** Permiten crear dominios complejos para almacenar datos en los atributos de las tablas.
- **Tablas Anidadas:** Permiten seleccionar valores dentro de una lista para un atributo. Representan relaciones uno a varios, donde el contenido de la tabla representa la tabla relacionada con la actual.
- **Varrays:** Es una colección de valores que permite seleccionar varios valores relacionados con un atributo de una tabla. Por ejemplo permitiría elegir varias asignaturas en el mismo alumno (lo que está en desacuerdo con el modelo relacional, pero es que en realidad la información del varray se almacena aparte)
- **Objetos Grandes:** Se trata de los llamados LOB (large object binary). Permiten almacenar textos larguísimos, archivos externos (de los que sólo se almacenará la dirección) y datos binarios.
- **Vistas-Objeto:** Permiten crear objetos a partir de las tablas de datos.

1.3.7 Historia de Oracle

A través del tiempo Oracle 10g ha tenido un sinnúmero de versiones que le han permitido posicionarse entre una de las mejores bases de datos en el mercado mundial.

Surge a finales de los 70 con el nombre de "Relational Software" como un estudio de George Koch sobre sistemas gestores de base de datos ya que usaba la filosofía de las bases de datos relacionales.

1979 cambia el nombre a Relational Software, Inc. y presenta su nuevo producto *Oracle 2.0* como una versión comercial de un sistema de administración de bases de datos. Esta versión no soportaba transacciones, pero sí toda la funcionalidad SQL de queries y joins.

En 1983 RSI cambia su nombre definitivo a Oracle Corporation, y lanza Oracle V3, agregando el manejo de transacciones a través de las instrucciones COMMIT y ROLLBACK. De hecho, el producto es decodificado en C lo que permite expandir las plataformas de ejecución para incluir los entornos Unix, cuando hasta aquí era solo sobre Digital VAX/VMS.

En 1984, Oracle V4 soporta consistencia de lectura y en 1985 Oracle V5 empieza a soportar el modelo Cliente/Servidor para unirse al auge de la aparición de las redes. Que además soporta la ejecución de queries distribuidos.

1989 trajo la aparición del ERP de Oracle, conocido como Oracle Financials ®, junto a la versión 6 del motor, que agrega un lenguaje procedural (PL/SQL), locking a nivel de fila y la posibilidades de hacer back up sin la necesidad de bajar los procesos.

Para convertirse en una base de datos completa, en 1992 aparece Oracle V7h, donde la h viene de datawareHouse, aunque lo más significativo es el soporte de la integridad referencial, el almacenamiento y ejecución de programas escritos en PL/SQL dentro del motor y la definición de disparadores de base de datos.

En 1997 Oracle V8 comienza a soportar desarrollos orientados a objetos y el almacenamiento y ejecución de contenido multimedia y en 1999 sale a la luz Oracle 8i para estar a tono con los requerimientos de la Internet, de donde se derive la i del nombre. Además, el motor incorpora una Java Virtual Machine interna para soportar el almacenamiento y ejecución de código Java dentro del motor.

En 2001 Oracle 9i trae más de 400 nuevas características incluyendo la habilidad de manipular documentos XML, opciones de alta disponibilidad, bases de datos en Clúster. Un avance importante se hace sobre la definición de Bases de Datos Virtuales (VPD), autenticación vía LDAP y en la autoadministración de la base de datos.

En 2003 Oracle Corporation lanza Oracle 10g, donde la "g" viene de "Grid", incorporando el manejo y administración de bases de datos malladas, un conjunto de bases de datos cuya administración de espacio, recursos y servicios pueden administrarse como si fueran una sola.

2004: Oracle compra Peoplesoft por 10.500 millones de dólares.

2005: Oracle adquiere Siebel Systems por 5.850 millones de dólares.

En el 2007, Oracle anunció la última versión de su Base de Datos denomina Oracle 11g.

2008: Oracle Adquiere BEA Systems por 8.500 millones de dólares y adquiere RuleBurst Holdings Limited, la empresa matriz de HaleyLimited

2009: El 20 de abril de 2009 se anuncia la adquisición de Sun Microsystems, en una operación que ronda los 7.400 millones de dólares.

2011: En Abril de 2011, Oracle adquiere la empresa británica Datanomic. En Junio de 2011 Oracle publica en su página web la compra de Pillar Data Systems y la empresa FatWire Software, especializada en software empresarial. En Julio de 2011, también adquiere la empresa KSpliceIncyInquire. El 24 de octubre de 2011 Oracle anuncia la adquisición de la compañía de cloud computing Right Now Technologies por 1.500 millones de dólares.

2012: En Febrero de 2012 Oracle compra la firma de gestión software Taleo por 1.900 millones dólares. En Abril 2012 compra Clear Trial. En Mayo 2012 Oracle ha comprado Vitruve, una firma de marketing social basada en el cloud, La operación está valorada en unos 239 millones de dólares. En Junio de 2012 Oracle anuncia la adquisición de Collective Intellect, que ofrece una aplicación de inteligencia social basada en el cloud^[W5].

1.3.8 Características

- Escalabilidad de departamentos a sitios empresariales.
- Robustez, Confiabilidad, disponibilidad y arquitecturas seguras.
- Un modelo de desarrollo de fácil despliegue.
- Plataforma Oracle (Incluyendo SQL, PL/SQL, Java y XML)
- Un manejo de interfaz para todas las aplicaciones.
- Estándares tecnológicos para las industrias, e interconectividad.

1.3.9 Requerimientos de Hardware y Software

Los requerimientos que Oracle recomienda para su instalación:

Requerimientos del Sistema	Mínimo	Recomendado
Memoria RAM	512 MB	1 GB
Almacenamiento SWAP	1 GB	2 GB
Almacenamiento Temporal	400 MB	1 GB
Espacio en Disco	1.5 GB Software 1.5 GB Base de Datos	1.5 GB Software 1.5 GB Base de Datos

Tabla 1.3. Requerimientos de Hardware y Software.
Fuente: [Propia].

^[W5]WIKIPEDIA.(s.f.). Oracle Corporation. Recuperado el 24 de 12 del 2012, de http://es.wikipedia.org/wiki/Oracle_Corporation

1.3.10 Otros Servidores

Los rivales de Oracle fundamentales hoy en día son:

- Ingreso de Computer Associates.
- DB/2 de IBM.
- SQL Server de Microsoft.
- ADABAS de Software AG.
- Informix.
- PostgreSQL (de código abierto).

1.3.11 Estrategia de Oracle

Oracle es una empresa que fabrica diversos productos. Su idea es aportar con un producto autosuficiente para el mantenimiento de datos y la creación de aplicaciones basadas en estos.

Sus tres productos más importantes son:

- **Oracle DataBase.** El DBMS Oracle, junto con las herramientas fundamentales para hacer de servidor y los programas clientes necesarios para conectar clientes.
- **Oracle Application Server.** Es un contenedor de objetos rápido, ligero, escalable y de fácil uso, compatible y certificado con la Java 2 Enterprise Edition (J2EE). OC4J es 100% Java y se ejecuta bajo entornos estándar de Java Development Kit (JDK). Estas características permiten que OC4J se ejecute sobre cualquier plataforma que posea Máquinas Virtuales Java, sin necesidad de instalar distintas versiones del software.
- **Oracle Developer Suite.** Programas para la generación de aplicaciones rápidas basadas en bases de datos Oracle.

1.3.12 Servidor Oracle

Un servidor Oracle es el software que permite una administración y desarrollo de bases de datos y sus tres posibilidades de ejecución pueden ser:

- **Local o Basada en Host.** El servidor se ejecuta en la misma máquina en la que se conectan los clientes. La versión personal de Oracle data base, produce servidores de este tipo.
- **Cliente-Servidor.** Enfoque más típico. El servidor reside en un ordenador distinto respecto al que los usuarios van a usar para conectarse a la base de datos.

- **Cliente-Servidor de Aplicaciones-Servidor.** Los usuarios acceden a un servidor de aplicaciones que, a su vez, accede al servidor Oracle. Los tres elementos (cliente, servidor de aplicaciones, servidor Oracle) pueden estar en tres máquinas distintas.

1.3.13 Elementos del Servidor Oracle

El servidor Oracle está formado por dos elementos:

- **La Instancia de la Base de Datos:** Consta de **datos** (llamados estructuras de memoria) y de **procesos** en memoria (procesos background) necesarios para dar servicio a los usuarios de la base de datos. Puede haber más de una instancia si se distribuye la base de datos en más de una máquina. Cada instancia abre una y sólo una base de datos.
- **Ficheros en Disco:** Representan la base de datos en sí. Consta de **Estructuras lógicas** (tablespaces y objetos del esquema de usuario) y **Estructuras físicas** que son ficheros de datos almacenados en disco. Los ficheros de datos (asociados a los tablespaces), los ficheros redo logy los ficheros de control.

1.3.14 Conexiones

Para establecer una sesión con la base de datos, el usuario necesita de una herramienta cliente como SQL*Plus o ejecutar una aplicación de desarrollo (Oracle Forms); entonces se ejecuta un proceso de usuario y cuando esto ocurre, en el servidor se establece un proceso de servidor. Este proceso es el encargado de comunicar al usuario con la instancia Oracle. Cada vez que el usuario ejecuta instrucciones SQL¹⁴, éstas son transmitidas a la instancia Oracle por el proceso servidor. Cada sesión es una conexión de un usuario con el servidor Oracle. Un usuario puede establecer múltiples sesiones (si se conecta desde diferentes herramientas y máquinas).

1.3.15 Estructura de la Base de Datos Oracle

La base de datos Oracle está estructurada de la siguiente manera:

- **Archivos de Datos:** Contiene datos actuales de la base de datos y el diccionario de datos.
- **Archivos Rehacer (redo logs):** Almacenan datos recuperables en caso de error grave.
- **Archivos de Control:** Necesarios para mantener la integridad de la base de datos.

Además se utilizan otros archivos de forma auxiliar:

¹⁴El lenguaje de consulta más popular de los sistemas de base de datos relacionales.

- **Archivos de Parámetros:** Que definen algunas características de una instancia Oracle.
- **Archivos de Contraseñas:** Que sirven para autenticar a los usuarios.
- **Copias de Archivos Rehacer:** Utilizadas para la recuperación de datos.

1.3.16 Instancia de la Base de Datos

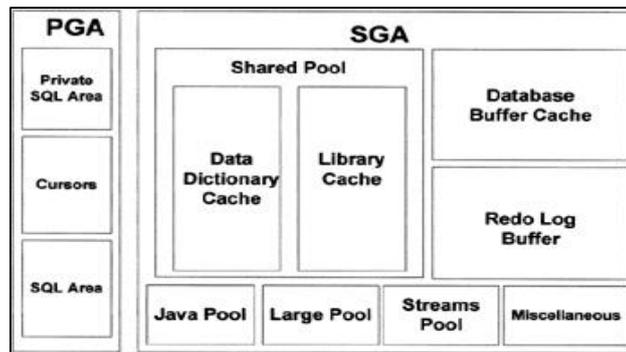


Figura 1.15. Instancia de la Base de Datos.
Fuente: [W6].

La instancia de la base de datos sirve para gestionar la base de datos y proporcionar servicio a los usuarios que acceden a la misma, esta consta de estructuras en memoria y procesos en segundo plano (background).

➤ Estructuras en Memoria

SGA.

Es la abreviatura de System Global Área, situada al inicio de los datos de la instancia y contiene los datos e información de control de la instancia. Está formada por las siguientes estructuras:

- **Shared Pool, Fondo Común Compartido:** Almacena las últimas instrucciones SQL y PL/SQL ejecutadas. Posee dos estructuras internas:
 - ✓ *Caché de instrucciones (Library cache):* Almacena las últimas instrucciones SQL y PL/SQL ejecutadas. Administra los datos mediante algoritmo LRU.
 - ✓ *Caché del diccionario de datos:* Almacena las últimas definiciones de la base de datos utilizadas (tablas, índices, privilegios, usuarios,...) Cada vez que una instrucción utiliza un nombre de la base de datos (tabla, índice,...) se comprueba en

[W6] ANÓNIMO. (s.f.). Oracle Components. Recuperado el 14 de Diciembre del 2012, de <http://flylib.com/books/en/3.396.1.23/1/>

el diccionario de datos y se almacena en este caché. De este modo la siguiente vez no hace falta acceder al diccionario de datos real.

- **Caché Buffer:** Almacena los últimos bloques de datos accedidos por los usuarios.
- **Buffer de Archivo Rehacer:** Almacena los últimos cambios realizados a los bloques de datos de la base de datos.
- **Large Pool:** Se utiliza como memoria de sesión y para realizar operaciones de backup.
- **Java Pool:** Se utiliza como caché de los comandos Java.

PGA

Zona global de los programas (Program Global Area). En ella se almacenan los datos correspondientes a un proceso (sólo un proceso puede utilizar esta área). Incluye:

- **Áreas de Ordenación:** Para acelerar las tareas de ordenación de datos.
- **Información de Sesión:** Usuario, privilegios, etc.
- **Estado del Cursor:** Tareas SQL actualmente en ejecución.
- **Espacio de Pila:** Variables y otros datos.

➤ Procesos en Segundo Plano

En Oracle los procesos pueden ser de estos tipos:

- **Proceso de Usuario:** Lanzado por el usuario para pedir interacción con la base de datos.
- **Proceso de Servidor:** Hacen de enlace entre los procesos de usuarios y el servidor Oracle. Se utilizan como manejadores de los procesos de usuario. Los comandos de usuario se envían a estos procesos que se encargan de solicitar peticiones a la base de datos mediante el interfaz de programas de Oracle (OPI, Oracle Program Interface).
- **Procesos en Segundo Plano (background).** Cada instancia de Oracle arranca una serie de procesos background. Los procesos obligatorios son:
 - ✓ *DBWR (DataBase Writer):* Escribe en los ficheros de datos los buffers más antiguos de la memoria, para que la base de datos vaya almacenando los cambios.
 - ✓ *LGWR (LoG Writer):* Escribe los datos a los ficheros rehacer (redo) desde la caché de archivos rehacer.

- ✓ *CKPT*: Actualiza todas las cabeceras de los ficheros de datos para que aparezca la nueva disposición de datos. Esto ocurre cuando se genera un punto de comprobación.
- ✓ *SMON (System Monitor)*: Permite recuperar la instancia de la base de datos en caso de caída fatal (cuando el sistema falla por ejemplo). Cuando se reinicia de nuevo la instancia de la base de datos.
- ✓ *PMON (Process Monitor)*: Es el encargado de gestionar adecuadamente los procesos que fallan. Ante caídas de procesos, PMON se encarga de restaurarlos datos adecuadamente.
- ✓ *SQL *Net Listener*: Es el encargado de encaminar por una red solicitudes de un cliente a un servidor de base de datos Oracle. Este proceso escuchador (listener) está tanto en el cliente como en el servidor. Puede encaminar solicitudes que se dirigen a varias instancias.

1.3.17 Procesamiento de Instrucciones SQL

Para poder ejecutar SQL sobre la base de datos, hay que conectar con la instancia Oracle de la base de datos, lo cual requiere la comunicación entre un proceso cliente y el servidor (el proceso cliente puede ser una instancia de SQL*Plus por ejemplo). Los componentes utilizados por Oracle para procesar el SQL dependen del código enviado:

De manera general los pasos en ese proceso son:

- El usuario abre la herramienta que permite el envío de peticiones SQL (SQL*Plus).
- El usuario introduce su nombre de usuario y contraseña.
- Oracle consulta el diccionario de datos para verificar la existencia del usuario y para validar su permiso de conexión. Si lo tiene, se produce la conexión
- El usuario escribe la instrucción SQL (modificación).
- Oracle traduce la instrucción con el analizador de instrucciones (devolvería un error si la instrucción no es válida).
- Oracle traduce los nombres usados en la instrucción con la ayuda del diccionario de datos.
- Si es una instrucción SELECT, comprueba si otros usuarios han enviado la misma instrucción con la caché de instrucciones de la SGA. Si la instrucción está ahí coge los resultados del buffer caché de la base de datos.
- Si la instrucción conlleva cambios, el servidor bloquea las filas que se modificarán.

- La base de datos graba los cambios (si les hubo) y actualiza los archivos deshacer.
- La base de datos graba los nuevos valores para los datos.
- Oracle libera del bloqueo los registros.
- El usuario recibe un mensaje de éxito.

1.3.18 Almacenamiento

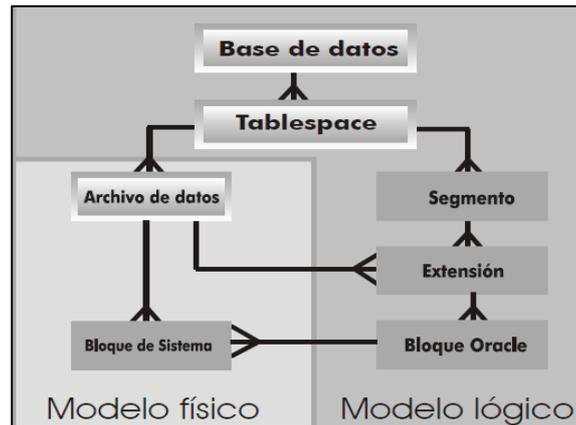


Figura 1.16. Almacenamiento.
Fuente: [D5].

Una base de datos tiene una estructura lógica (que se manipula mediante comandos) y una estructura física (la que realmente se almacena en disco).

➤ Estructura Lógica

- **Tablespaces:** Pertenecen sólo a una base de datos y sirven para agrupar los datos de la base de datos. Cada tablespace está formado físicamente por uno o más archivos de datos. Están divididos en 0 o más segmentos. Se pueden visualizar en línea o fuera de línea y pueden ser activados en sólo lectura o en lectura / escritura.

[D5] Sánchez, J. (2004). Arquitectura del DBMS de Oracle. Recuperado el 18 de Septiembre del 2012, de <http://www.jorgesanchez.net/bd/arquOracle.pdf>

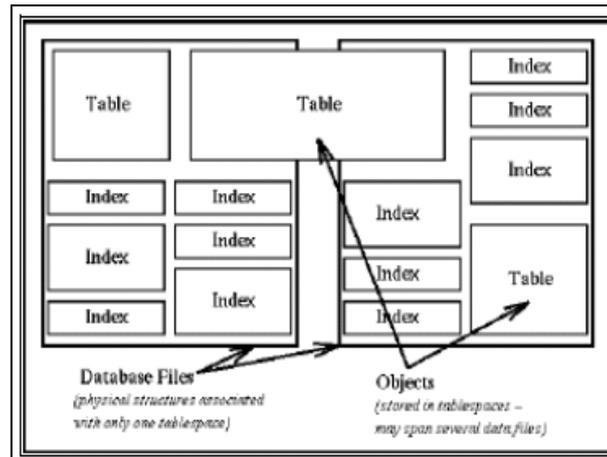


Figura 1.17. Estructura Lógica.

Fuente: ^[D9].

- **Segmento:** Sirven para almacenar las estructuras lógicas de la base de datos (tablas, índices, etc.).
- **Extensiones:** División que se hace a cada segmento. El DBA puede añadir o quitar extensiones a los segmentos a fin de hacer que ganen o pierdan espacio.
- **Bloque Oracle o Bloque de Datos:** Es la unidad mínima de datos y se corresponde a una o más unidades de datos mínimas del sistema operativo en el que nos encontremos.

➤ Estructura Física

- **Archivos de Datos:** Son archivos en disco que sirven para almacenar los datos físicamente. Cada archivo de datos pertenece sólo a un tablespace. Su tamaño se puede gestionar.
- **Bloques de Sistema:** La división mínima de los datos que hace el sistema operativo.

1.3.19 Transacciones

Una transacción son varias operaciones SQL que forman una unidad de trabajo, comienza cuando una persona se conecta y ejecuta la instrucción *commit* o anularla ejecutando la instrucción *rollback*, la anulación deja la base de datos en el estado anterior al comienzo de la transacción. En Oracle se admite además el uso de puntos de ruptura *checkpoints* para almacenar valores intermedios y volver a cualquier de ellos si interesa. Pero esto ralentiza excesivamente el sistema.

^[D9]Oracle. (2008).Curso de Administración de Oracle 10g (10.2) Manual del Alumno. Recuperado el 21 de Julio del 2012, de <http://es.scribd.com/doc/19482492/Curso-de-Oracle-10g-Administracion-nivel-basico>

1.3.20 Usuarios.

Es una cuenta que permite a alguien conectarse a Oracle y según sus derechos de seguridad usar, crear, y/o manejar objetos de la base de datos. Con la cuenta de usuario se pueden definir:

- Nombre único del usuario y clave.
- Clave expirada (obligará el usuario a cambiar su clave al ingresar la primera vez).
- Account locked: cuenta bloqueada.
- Default tablespace: donde se guardarán por defecto sus objetos.
- Temporary tablespace.
- Roles (conjunto de derechos) y privilegios de sistema y de objetos.
- Quotas: espacio autorizado.
- Perfiles.

Los roles y privilegios se agregan después de haber creado el usuario.

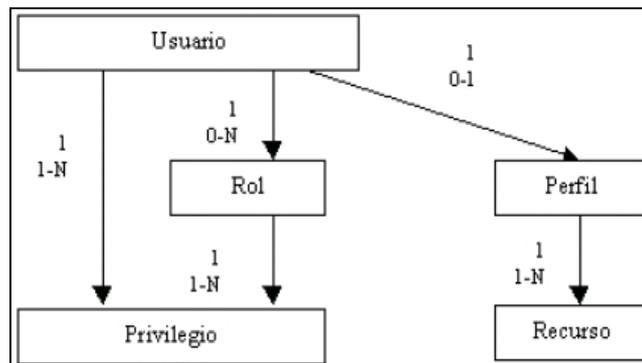


Figura 1.18. Privilegios de Usuario.

Fuente:^[D9].

Concepto	Significado
Privilegio	Permiso para realizar una acción, asignable a un usuario o un rol
Rol	Conjunto de privilegios, asignables a un usuario o un rol
Usuario	Colección de objetos y privilegios identificado con un nombre y password
Perfil	Conjunto de restricciones relativas al uso de recursos y asignable a usuarios. Un usuario solo puede tener un perfil

^[D9]Oracle. (2008).Curso de Administración de Oracle 10g (10.2) Manual del Alumno. Recuperado el 21 de Julio del 2012, de <http://es.scribd.com/doc/19482492/Curso-de-Oracle-10g-Administracion-nivel-basico>

Recurso	Uso susceptible de ser restringido, asignable a un perfil
---------	---

Tabla 1.4. Privilegios de Usuario.

Fuente: [Propia].

➤ Usuarios Especiales

- **SYS:** Usuario con acceso al esquema SYS que guarda las tablas y vistas para el diccionario de datos de la base de datos, se usa para cambiar los parámetros de almacenamiento del diccionario de datos y se crea automáticamente al crear una base de datos y tiene rol DBA.
- **SYSTEM:** Se usa para crear tablas y vistas con información relativa a la administración de la base de datos y se crea automáticamente al crear una base de datos y tiene rol DBA.

➤ Crear Cuentas de Usuario

La sintaxis para crear cuentas de usuario es la siguiente:

```
CREATE USER <nombre_usuario> IDENTIFIED {BY <password>} <opciones>;
```

Las opciones pueden ser:

```
DEFAULT TABLESPACE <nombre_tablespace>
TEMPORARY TABLESPACE <nombre_tablespace>
QUOTA INT {K | M} ON <nombre_tablespace>
QUOTA UNLIMITED <nombre_tablespace>
PROFILE <nombre_perfil>
PASSWORD EXPIRE
ACCOUNT {LOCK|UNLOCK}
```

Ejemplo de creación de una cuenta de usuario:

```
CREATE USER UTNDB IDENTIFIED BY "cristian"
DEFAULT TABLESPACE ACADEMICO
TEMPORARY TABLESPACE TEMP
PROFILE DEFAULT
ACCOUNT UNLOCK;
```

➤ Modificar una Cuenta de Usuario

Para modificar una cuenta de usuario se utiliza la siguiente sintaxis:

```
ALTER USER <nombre_usuario> <opciones>;
ALTER USER <nombre_usuario>,... {GRANT|REVOKE} <opciones_proxy>;
```

Las opciones pueden ser:

```
IDENTIFIED BY <password> [REPLACE <old_password>]
IDENTIFIED EXTERNALLY
DEFAULT TABLESPACE <nombre_tablespace>
TEMPORARY TABLESPACE <nombre_tablespace>
QUOTA INT {K | M} ON <nombre_tablespace>
QUOTA UNLIMITED ON <nombre_tablespace>
PROFILE <nombre_perfil>
DEFAULT ROLE <role> [, <role>,...]
DEFAULT ROLE ALL [EXCEPT [, <role>,...]]
DEFAULT ROLE NONE
PASSWORD EXPIRE
ACCOUNT {LOCK|UNLOCK}
```

Las opciones que puede tomar el proxy son:

```
CONNECT THROUGH <proxy> [WITH ROLE <nombre_rol>,...] [authentication]
CONNECT THROUGH <proxy> [WITH ROLE ALL EXCEPT <nombre_rol>,...] [authentication]
CONNECT THROUGH <proxy> [WITH NO ROLES] [authentication]
```

➤ Eliminar una Cuenta de Usuario

La sintaxis para eliminar una cuenta de usuario es la siguiente:

```
DROP USER <nombre_usuario> [CASCADE];
```

Se utiliza *CASCADE* para borrar todos los objetos que contiene el esquema del usuario. Ejemplo:

```
DROP USER UTNDB CASCADE;
```

1.3.21 Privilegios

Los privilegios pueden ser otorgados a nivel de sistema o a nivel de objetos.

➤ Privilegios de Sistema

Los roles de sistema se utilizan para distribuir la disponibilidad de los comandos del sistema utilizados para gestionar la BD.

Estos privilegios que pueden tener un alcance limitado al esquema del usuario o de todos los esquemas. La palabra clave *ANY* significa que ese usuario tiene el privilegio para todos los esquemas en la BD. Por ejemplo, el derecho de crear una sesión (conectarse), crear tablas, crear respaldos, etc.

➤ Privilegios de Objetos

Por defecto un usuario tiene todos los derechos sobre sus objetos (esquema). Un usuario puede otorgar (*GRANT*) privilegios sobre sus objetos a cualquier usuario o rol (role). Un privilegio puede ser otorgado de manera explícita a un usuario, o se puede dar a un role, y cualquier usuario con este rol tendrá este derecho.

Los privilegios que pueden otorgarse sobre objetos son los siguientes:

Privilegio	Capacidades Otorgadas
SELECT	Puede consultar a un objeto.
INSERT	Puede insertar filas en una tabla o vista. Puede especificarse las columnas donde se permite insertar dentro de la tabla o vista.
UPDATE	Puede actualizar filas en una tabla o vista. Puede especificarse las columnas donde se permite actualizar dentro de la tabla o vista.
DELETE	Puede borrar filas dentro de la tabla o vista.
ALTER	Puede alterar la tabla.
INDEX	Puede crear índices de una tabla.
REFERENCES	Puede crear claves ajenas que referencie a esta tabla.
EXECUTE	Puede ejecutar un procedimiento, paquete o función

Tabla 1.5. Privilegios de Objeto.

Fuente: [Propia].

Para ver los datos de una vista, se necesita el privilegio sobre la vista, y no sobre las tablas que contiene la vista. A nivel de una vista, es posible limitar los privilegios a ciertos campos, y a ciertos rangos de valores de las filas regresadas. Es una opción para datos muy sensible, pero es un poco pesado a establecer y mantener.

1.3.22 Otorgar/Quitar Privilegios

Los comandos *GRANT* y *REVOKE*, permiten otorgar o quitar respectivamente un privilegio.

El grupo *PUBLIC* es un privilegio otorgado al grupo 'public' que es otorgado a todos los usuarios de la base de datos.

Los derechos se pueden dar '*with grant option*', lo que significa que el usuario tendrá el derecho de otorgar este mismo derecho a otro(s) usuario(s).

Ejemplo de privilegios otorgados a un usuario sobre una tabla:

```
GRANT ALTER, DELETE, INSERT, SELECT, UPDATE, ON COMMIT REFRESH, QUERY
REWRITE, DEBUG, FLASHBACK ON RHC_TAB_ESTADO_RESOLUCION TO CSUAREZ;
```

Ahora se procede a quitar los privilegios otorgados a un usuario sobre una tabla:

```
REVOKE ALTER, DELETE, INSERT, SELECT, UPDATE, ON COMMIT REFRESH, QUERY  
REWRITE, DEBUG, FLASHBACK ON RHC_TAB_ESTADO_RESOLUCION TO CSUAREZ;
```

1.3.23 Roles

Son agrupaciones de privilegios que facilitan la tarea de gestionar a los usuarios. Así cuando una serie de usuarios van a tener los mismos privilegios, se crea un rol que contenga esos privilegios y a esos usuarios se les asigna el rol. Un rol puede otorgar privilegios sobre objetos o de sistema.

Oracle proporciona varios roles ya preparados, por ejemplo el rol DBA da privilegio absoluto a un usuario.

Para crear roles se utilizamos lo siguiente:

```
CREATE ROLE ADMINISTRADOR NOT IDENTIFIED;
```

Para eliminar roles utilizamos:

```
DROP ROLE ADMINISTRADOR;
```

➤ Esquemas

Los esquemas están asociados a los usuarios. Agrupan los objetos lógicos que pertenecen al usuario. Es decir es el conjunto de tablas, vistas, sinónimos, instantáneas, enlaces de base de datos, procedimientos y funciones, paquetes, etc. Cada usuario tiene su propio esquema y, en principio, un usuario no tiene acceso a los elementos de otro usuario, salvo que sea un administrador o que otro usuario ceda el privilegio de utilización de uno o más de sus objetos al resto de usuarios.

➤ Tablas

Las tablas son la unidad básica de almacenamiento de datos en Oracle, los datos son almacenados en filas y columnas. Se define a través de un nombre y un conjunto de columnas (nombre y tipo). Al crear una tabla Oracle asigna un segmento de datos en el tablespace. Se puede controlar el espacio y el uso de ese segmento.

Para **crear una tabla** tener en cuenta: el nombre de la tabla, el tipo de dato de cada columna, en que esquema y en que tablespace la vamos a crear. La sintaxis para crear tablas es la siguiente:

```
CREATE TABLE <nombre_esquema>.<nombre_tabla> (  
<nombre_columna> <tipo_dato> [<restriccion_columna>],  
...
```

```
) TABLESPACE <nombre_tablespace>;
```

El siguiente ejemplo es una muestra de cómo crear tablas:

```
CREATE TABLE UTNDB.MGI_TAB_ESTADO_PROYECTO  
(  
  ESTADO          VARCHAR2 (1 BYTE),  
  CODIGO          VARCHAR2 (10 BYTE) NOTNULL,  
  DESCRIPCION     VARCHAR2 (50 BYTE),  
  OBSERVACION     VARCHAR2 (100 BYTE)  
)  
TABLESPACE ACADEMICO;
```

A través de este ejemplo se identifica a *UTNDB* como el nombre del esquema. La sentencia *CREATE TABLE* permite crear la tabla denominada *MGI_TAB_ESTADO_PROYECTO*, la cual está conformada por las columnas *ESTADO*, *CODIGO*, *DESCRIPCION* y *OBSERVACION*. El nombre del tablespace se denomina *ACADEMICO*.

Para **borrar una tabla** se utiliza la siguiente sintaxis:

```
DROP TABLE <nombre_esquema>.<nombre_tabla> CASCADE CONSTRAINTS;
```

Ejemplo:

```
DROP TABLE UTNDB.MGI_TAB_ESTADO_PROYECTO CASCADE CONSTRAINTS;
```

La instrucción *DROP TABLE* borra la tabla *MGI_TAB_ESTADO_PROYECTO* que pertenece al esquema *UTNDB*, emplea la instrucción *CASCADE CONSTRAINTS* para borrar las referencias que existan de esta tabla a otras.

Para **modificar la definición de una tabla** se utiliza el comando *ALTER TABLE*, se puede añadir un nuevo atributo o modificar uno ya existente. Estas dos operaciones se pueden realizar teniendo en cuenta una serie de restricciones que posteriormente se detallan.

```
ALTER TABLE <nombre_esquema>.<nombre_tabla>  
ADD < nombre_columna> <tipo_dato> <restriccion_columna>  
...  
MODIFY <nombre_columna> <tipo_dato> <restriccion_columna>  
.... ;
```

La opción *ADD* permite añadir un nuevo atributo a la tabla con un valor *NULL* inicial para todas las filas. *MODIFY* cambia la definición del atributo que ya existe en la tabla. Las restricciones a tener en cuenta son las siguientes:

- Restricción columna sólo puede ser *NOT NULL*.
- Se puede cambiar el tipo del atributo o disminuir su tamaño si todas las filas tienen en ese atributo el valor *NULL*.
- Un atributo *NOT NULL* únicamente se puede añadir a una tabla sin filas.
- Un atributo ya existente sólo se puede hacer *NOT NULL* si todas las filas tienen en ese atributo un valor distinto de *NULL*.
- Cuando se modifica un atributo, todo lo no especificado en la modificación se mantiene.

Ejemplo:

```
ALTER TABLE UTNDB.MGI_TAB_ESTADO_PROYECTO  
ADD (PRUEBA VARCHAR2 (50 BYTE))  
MODIFY (OBSERVACION VARCHAR2 (100 BYTE));
```

La **habilitación o deshabilitación de restricciones de integridad** se define mediante el comando *CREATE TABLE* o se añade con el comando *ALTER TABLE*, la condición se habilita automáticamente. Una restricción puede ser posteriormente deshabilitada con el comando *ALTER TABLE*:

```
ALTER TABLE <nombre_tabla> DISABLE  
CONSTRAINT <nombre_constraint> PRIMARY KEY  
UNIQUE <lista_columnas> CASCADE;
```

Para deshabilitar una clave primaria, han de deshabilitarse previamente todas las restricciones de clave externa que dependan de esa clave primaria. La cláusula *CASCADE* deshabilita automáticamente las restricciones de clave externa que dependen de la (deshabilitada) clave primaria.

Ejemplo: Deshabilitar la clave primaria de la tabla *MGI_TAB_ESTADO_PROYECTO*:

```
ALTER TABLE MGI_TAB_ESTADO_PROYECTO DISABLE PRIMARYKEY CASCADE;
```

1.3.24 Pérdida de Información

Es una de las tareas y herramientas fundamentales que nos proporcionan las bases de datos. Hay posibilidad de perder datos de nuestra base de datos por alguna de estas razones:

- **Fallo en una Instrucción:** Hay instrucciones que pueden provocar la pérdida no deseada de cientos de registros en un momento (*DELETE* o *UPDATE*). En ese caso basta ejecutar un *rollback* antes de que la instrucción se lleve a cabo.

- **Fallo en la Comunicación:** Si una conexión de usuario se corta anulando el proceso de usuario relacionado. En ese caso, Oracle anula los cambios de la última transacción (el resto de transacciones sí se almacena).
- **Caída del Servidor:** Puede que la instancia Oracle se deje de ejecutar. En ese caso basta con lanzar de nuevo la instancia. El proceso SMON se encargará de grabar los archivos rehacer y aplica de nuevo las transacciones confirmadas. Se anulan los cambios no confirmados.
- **Pérdida de Datos en los Archivos:** Es el único caso en el que tiene que intervenir el administrador. La única posibilidad es recuperarles de una copia de seguridad.

1.3.25 Copias de Seguridad (Backups)

Es una de las herramientas fundamentales de toda base de datos. Al hacer la copia de seguridad se nos permite recuperar la información de esa copia. Eso hace que perdamos definitivamente los datos perdidos desde la última copia de seguridad, de ahí la importancia de hacer copia a menudo.

Hay una posibilidad de perder menos información y es hacer que la base de datos se ejecute en modo Archivo Log, lo que significa que se almacena en un archivo de datos especial, los datos que se van rechazando en los registros rehacer por estar llenos. Lo malo de esta opción, es que el servidor funciona más lento; lo bueno es que en caso de desastre se pueden recuperar los datos almacenados.

“Los *backups* se pueden clasificar en físicos y lógicos. Los físicos se realizan cuando se copian los ficheros que soportan la BD. Entre estos se encuentran los *backups* del SO, los *backups* en frío y los *backups* en caliente.

Los *backups* lógicos sólo extraen los datos de las tablas utilizando comandos SQL y se realizan con la utilidad export/import.”^[W7]

➤ Físicos

Copian físicamente los datos de la BD.

- **Del Sistema Operativo:** Consume mucho tiempo y se hace inaccesible al sistema mientras se lleva a cabo. Aprovecha el backup del SO para hacer el de la BD.

^[W7]Jesús, V. (Mayo de 1998). Oracle Backup y Recuperación. Recuperado el 15 Septiembre del 2012, de <http://www.infor.uva.es/~jvegas/cursos/bd/oraback/oraback.html>

- **En Frío:** Hay que parar las BD en modo normal y copiar los ficheros sobre los que se asienta. Una vez realizada la copia se puede volver a arrancar.
- **En Caliente:** Se hace la copia mientras la BD está abierta y funcionando en modo ARCHIVELOG. Consiste en copiar todos los ficheros correspondientes a un tablespace determinado, para todos los tablespace de la BD.

➤ Lógicos

Los backups lógicos se hacen con la herramienta *export* que copia los datos y la definición de la BD en un fichero en un formato interno de Oracle. Copian el contenido de la BD pero sin almacenar la posición física de los datos.

Esta utilidad permite al administrador de la base de datos hacer copias de determinados objetos de la BD, así como restaurarlos o moverlos de una BD a otra.

1.3.26 Herramientas de Oracle

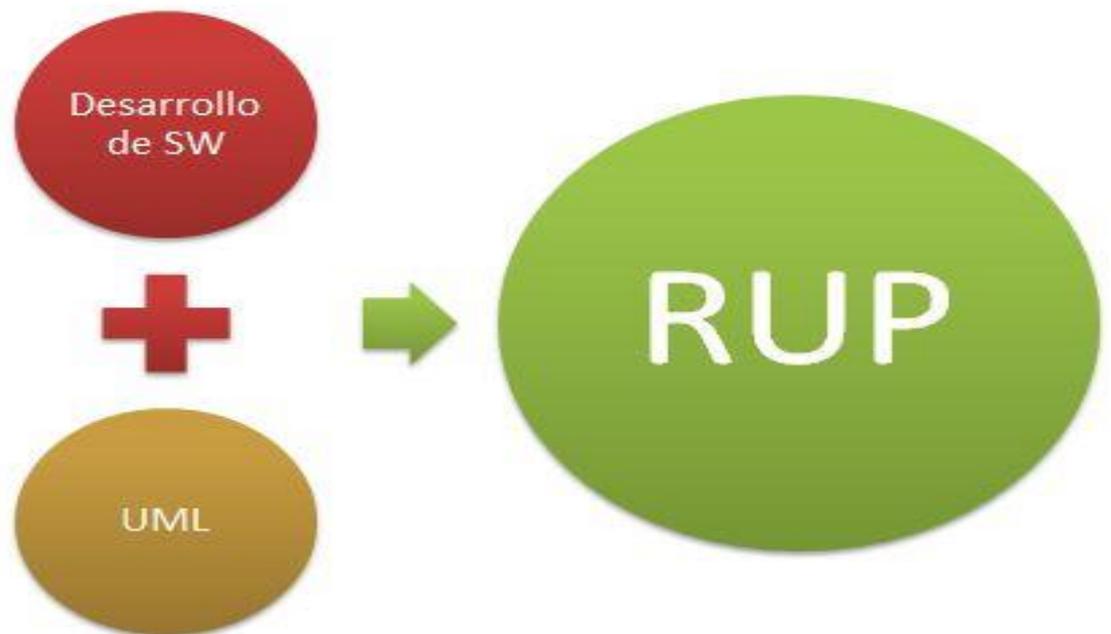
El software del sistema de bases de datos Oracle incorpora herramientas para realizar la mayoría de tareas comunes en una base de datos:

- **Oracle Universal Installer:** Controla cada nueva instalación de software Oracle a fin de que se integren de la mejor manera posible.
- **SQL*plus:** Programa cliente que permite conexión con el servidor Oracle para enviarle secuencias SQL y PL/SQL.
- **iSQL*plus:** Permite conexiones al servidor Oracle con la misma finalidad que el anterior pero utilizando un navegador de Internet, lo que facilita el trabajo.
- **SQL*plus WorkSheet:** Permite conexiones al servidor de Oracle, utilizando un entorno más potente (procede del Oracle Enterprise Manager)
- **Oracle Enterprise Manager:** Entorno que permite la administración y configuración completa del servidor Oracle.
- **SQL*Loader:** Permite cargar en bases de datos de Oracle información que procede de un archivo de texto. Necesaria para utilizar en las bases de datos de Oracle, información que procede de otro software.
- **Import/Export:** Para importar y exportar datos entre instancias de Oracle. De un servidor a otro por ejemplo. También se utiliza como herramienta de copia de seguridad.

- **Servidor Http de Oracle:** Basado en el servidor Apache, permite opciones de documentación y sobre todo la comunicación directa a través de iSQL*Plus con el servidor Oracle sin tener necesidad de instalar software adicional.
- **Net Manager:** Permite la administración de los servicios de red a fin de configurar las conexiones hacia instancias de Oracle.
- **Oracle Forms:** Permite crear aplicaciones visuales sobre bases de datos de Oracle.
- **Oracle Reports:** Asistente para la producción de informes.
- **Oracle Designer:** Herramienta CASE de Oracle, para crear esquemas en el ordenador y que el software produzca los resultados del mismo.
- **Oracle JDeveloper:** Crea aplicaciones Java pensadas para desarrollar formularios sobre datos de Oracle.
- **Oracle Developer Suite:** Integra todos los componentes anteriores.
- **Oracle AS (Application Server):** Servidor de aplicaciones de Oracle. Permite compilar aplicaciones J2EE.
- **Pro C/C++ Precompilador de C/C++ para Oracle**

CAPÍTULO II

DESARROLLO DEL PROYECTO



SISTEMA DE GESTIÓN DE
LA INVESTIGACIÓN
CIENTÍFICA, TECNOLÓGICA
E INNOVATIVA PARA EL
CUICYT - UTN

2.1 DOCUMENTO DE VISIÓN

2.1.1 Propósito

El propósito de este documento es definir a alto nivel los requerimientos del “Sistema para la gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN”.

Este sistema se encarga de administrar los proyectos de investigación presentados, lo mismo que pasan por diferentes etapas desde la inscripción hasta la finalización del mismo; las etapas por las que pasa un proyecto de investigación son las siguientes: presentación, evaluación, aprobación, oficialización, ejecución y cierre.

El sistema realiza tareas como el registro de investigadores, registro del currículum de los investigadores, creación de convocatorias (Proyectos Internos, Proyectos Externos, Proyectos de Aula, Tesis de Grado) de acuerdo a determinadas bases previamente definidas por el Consejo Directivo del CUICYT, registro de evaluadores de proyectos en una convocatoria realizada, asignación de proyectos a los evaluadores para que evalúen el mismo, registro de proyectos de investigación con su respectiva planificación técnica y económica, evaluación de los proyectos presentados empleando una matriz de evaluación, aprobación de los proyectos que cumplan los lineamientos establecidos, oficialización de los proyectos aprobados cargándolos al Plan Operativo Anual (POA) de la dependencia, ejecución de los proyectos los mismos que serán monitoreados bajo la creación de informes (técnico y económico) y cierre de los proyectos bajo la creación de un informe final que será aprobado por el Consejo Directivo del CUICYT.

El detalle de cómo el sistema cubre las necesidades de los usuarios se especifica en los casos de uso, que son información adicional no especificada en este documento.

➤ Alcance

Este documento de visión se aplica al “Sistema para la gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN”.

El cual será implementado y desarrollado por el tesista Cristian Herrera, con la dirección del Ing. Fernando Garrido. Director del departamento de Informática de la Universidad Técnica del Norte.

➤ Definiciones, Siglas y Abreviaturas

- Ver Glosario.

➤ **Referencias**

- Glosario: Definiciones abreviaturas y siglas.
- Resumen de los requerimientos de los interesados.
- Resumen del modelo de casos de uso.

2.1.2 Posicionamiento

➤ **Oportunidad de Negocio**

Actualmente la Universidad Técnica del Norte está luchando por la acreditación internacional, por lo que a través del CUICYT ha impulsado la generación y desarrollo de nuevos conocimientos especialmente con la ejecución de proyectos de investigación.

Con el objetivo de mantener un control permanente de estos proyectos el CUICYT ha visto la necesidad de disponer de un sistema informático que ayude a mantener una base de datos consistente con información íntegra, confiable y disponible en todo momento.

➤ **Definición del Problema**

El problema de	Administrar los proyectos de investigación que se ejecutan el CUICYT. Gestionar un buen seguimiento técnico y económico de los proyectos.
Que afecta a	Toda la comunidad Universitaria al no saber cuáles son los montos y el estado de los proyectos ejecutados en el CUICYT. A autoridades del CUICYT al no contar con una base de información oportuna para el levantamiento de información.
El impacto de ello es	El CUICYT realiza procesos lentos para el control de proyectos porque no cuenta con una aplicación informática que permita agilizarlos. Los proyectos no se encuentran ingresados en una base de datos por lo que la comunidad universitaria no conoce de ellos.
Una solución exitosa debería	Implementar un sistema informático de calidad soportada por una metodología eficiente de desarrollo de software, la cual permitirá administrar los proyectos. Este sistema además de agilizar los procesos para realizar una buena toma de decisiones, pretende ser una base de datos documental, ya que almacena información efectiva de los proyectos.

Tabla 2.1. Definición del Problema.

Fuente: [Propia].

➤ **Sentencia que define la posición del producto**

Para	El Centro Universitario de Investigación Científica y Tecnológica de la UTN.
Quienes	Director, secretaria, consejo directivo, investigadores, docentes, estudiantes, asistentes de investigación, digitador, analista económico,
El nombre del producto	"Sistema de gestión de la investigación Científica, Tecnológica e Innovativa del CUICYT-UTN".
Qué	Almacena la información de todos los proyectos de investigación.
No como	La forma manual que actualmente se está realizando.
Nuestro producto	Permite automatizar los procesos realizados con respecto a la gestión de la investigación, lo que implica un correcto registro, control y resguardo de la información; esto se lo realiza mediante una interfaz gráfica, sencilla y amigable. Además proporciona un acceso rápido y actualizado a la información desde cualquier lugar de la UTN.

Tabla 2.2. Definición de la Posición del Producto.

Fuente: [Propia].

2.1.3 Descripción de los Interesados y Usuarios

➤ **Resumen de los Interesados**

Interesados son todas aquellas personas directamente involucradas en la definición y alcance del proyecto. A continuación se presenta la lista de los interesados:

Nombre	Descripción	Responsabilidad
Coordinador del proyecto	Responsable a nivel directivo de la UTN del proyecto.	Establecer los lineamientos generales para el desarrollo del proyecto. Coordinar a nivel directivo los diferentes requerimientos que surjan en el desarrollo del sistema.
Responsables del proyecto	Responsable del proyecto por parte del Departamento de Informática de la UTN.	Responsable del análisis y diseño del proyecto. Gestiona el correcto desarrollo del proyecto en la construcción e implantación.
Responsables funcional	Responsable del proyecto por parte del CUICYT.	Responsable de coordinar con los diferentes usuarios la correcta

		determinación de los requerimientos y la correcta concepción del sistema.
Director, secretaria, asistente de investigación, analista económico y digitador	Responsable de la información para administrar los proyectos.	Responsable de definir la información que se utilizará para generar el proceso investigativo.

Tabla 2.3. Descripción de los interesados.

Fuente: [Propia].

➤ Resumen de los Usuarios

Los usuarios son todas aquellas personas involucradas directamente en el uso del Sistema de Gestión de la Investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN. A continuación se presenta una lista de los usuarios:

Nombre	Descripción	Responsabilidad
Administrador del sistema	Persona del Departamento de Informática que administra el "Módulo para la gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN".	Administrar eficazmente el sistema (gestionar acceso a usuarios, facilitar mantenimiento al sistema frente a nuevos requerimientos).
Administrador funcional del sistema	Persona del CUICYT de la Universidad Técnica del Norte que administra el "Módulo para la gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN".	Administrar funcionalmente el módulo, con nuevos cambios y nuevas necesidades.
Usuario del módulo	Director del CUICYT, Secretaria, investigadores, directores, estudiantes, asistentes de investigación.	Ingresar la información necesaria al sistema y administrarlo correctamente.

Tabla 2.4. Resumen de Usuarios.

Fuente: [Propia].

➤ **Entorno de Usuario**

El “Sistema de Gestión de la investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN”, beneficia al CUICYT y a toda la comunidad Universitaria, ya que permite mantener un control de las convocatorias realizadas con sus respectivos lineamientos, evaluadores de proyectos, proyectos asignados a los evaluadores, currículum vitae de los investigadores, de los proyectos, publicaciones de artículos científicos realizadas por los docentes e investigadores y eventos (capacitación, seminario, taller, foros, etc.) realizados con respecto a un determinado proyecto.

El “Sistema de Gestión de la investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN” interactúa con el sistema académico, sistema administrativo, sistema financiero y sistema de auditoría y seguridades.

2.1.4 Perfiles de los Stakeholders

➤ **Coordinador del Proyecto**

Representante	Ing. Fernando Garrido
Descripción	Responsable a nivel directivo del proyecto.
Tipo	Director
Responsabilidades	Establecer los lineamientos generales para el desarrollo del proyecto. Coordinar a nivel directivo los diferentes requerimientos que surjan en el desarrollo del sistema.
Criterios de éxito	Mantener una funcionalidad integral en los módulos. Mantener activa la aplicación luego de ser implantada.
Implicación	Revisor de la administración (Management Reviewer)
Entregables	N/A
Comentarios	Mantener una relación constante con el desarrollo del proyecto. Brindar apoyo a nivel gerencial cuando sea necesario.

Tabla 2.5. Perfil del Coordinador del Proyecto.

Fuente: [Propia].

➤ **Responsable del Proyecto**

Representante	Ing. Juan Carlos García
Descripción	Responsable del proyecto por parte del Departamento de Informática de la UTN
Tipo	Analista de sistemas
Responsabilidades	Responsable del análisis y diseño del proyecto. Gestiona el correcto desarrollo del proyecto en lo referente a la construcción e implantación.
Criterios de éxito	Cumplir con el cronograma determinado. Obtener un módulo de calidad que cumpla con los requerimientos funcionales establecidos.
Implicación	Jefe de proyectos (Project Manager).
Entregables	Ninguno. Funcionales establecidos.

Tabla 2.6. Perfil Responsable del Proyecto.
Fuente: [Propia].

➤ **Responsable Funcional.**

Representante	Dr. Marcelo Dávalos.
Descripción	Responsable del proyecto por parte del CUICYT.
Tipo	Experto en el tema
Responsabilidades	Responsable de coordinar con los usuarios la correcta determinación de los requerimientos y la correcta concepción del sistema. Coordinar las pruebas de validación del nuevo sistema. Coordinar y asegurar la capacitación de los usuarios.
Criterios de éxito	Obtener un sistema de calidad que cumpla con los requerimientos funcionales establecidos.
Implicación	Aprueba las especificaciones funcionales y las pruebas realizadas.
Entregables	Documento de revisión de las especificaciones funcionales. Documento de revisión de las pruebas funcionales.

Tabla 2.7. Perfil Responsable del Proyecto.
Fuente: [Propia].

➤ **Responsable del Desarrollo**

Representante	Sr. Cristian Herrera.
Descripción	Responsable del desarrollo del "Sistema de Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT.
Tipo	Programador.
Responsabilidades	Responsable del análisis y diseño del proyecto. Responsable del desarrollo del sistema.
Criterios de éxito	Obtener un sistema que cumpla con los requerimientos establecidos.
Implicación	Desarrollador de software.
Entregables	Documento de visión. Glosario. Lista de riesgos. Resumen del modelo de casos de uso. Especificaciones del modelo de casos de uso. Especificaciones complementarias.

Tabla 2.8. Perfil Responsable del desarrollo del Sistema.
Fuente: [Propia].

➤ **Usuario del Sistema.**

Representante	Centro Universitario de Investigación, Ciencia y Tecnología de la UTN.
Descripción	Director, Secretaria, Consejo Directivo, Investigadores, Estudiantes, Docentes y Asistentes de investigación.
Tipo	Usuario
Responsabilidades	Responsables de dar continuidad a los procesos de la investigación. Responsables de mantener actualizada la información.
Criterios de éxito	Sistema en funcionamiento. Obtener información íntegra, confiable y disponible en cualquier momento.
Implicación	N/A
Participación	Activa.

Tabla 2.9. Perfil de Usuarios.
Fuente: [Propia].

2.1.5 Necesidades de los Interesados y Usuarios

Necesidades	Prioridad	Inquietudes	Solución Actual	Solución propuesta
Diseñar un sistema que permita administrar los proyectos de investigación.	Alta	El sistema debe consolidar la información para facilitar la administración de los proyectos,	No existe una base de datos donde se pueda ingresar y obtener información de los proyectos.	Diseñar un sistema de calidad, para solucionar estas necesidades y brindar un mejor servicio.
Elaborar el sistema utilizando herramientas que facilite y agilice su desarrollo.	Alta	Se debe utilizar las herramientas existentes.	N/A	Desarrollar el sistema utilizando la herramienta Oracle Forms® 10g.
La interfaz del sistema debe ser fácil de manejar, cumpliendo con todos los requerimientos establecidos.	Alta	Cumplir con todos los requerimientos de los usuarios.	Desarrollo con la ayuda de los expertos del CUICYT.	Desarrollo con la ayuda de los expertos en el tema.
Mantener un registro premamamente de la gestión investigativa	Alta	Tomar en cuenta los datos necesarios a registrarse.	Estos datos se registran indistintamente en documentos de Word y de Excel.	Automatizar este proceso para garantizar la operatividad del sistema.

Realizar un sistema que sea adaptable a nuevos cambios.	Alta	N/A.	Los diferentes módulos que integran el ERP de la UTN son flexibles a nuevos cambios.	Diseñar un sistema que cumpla con las características de los demás módulos del ERP de la UTN.
---	------	------	--	---

Tabla 2.10. Necesidades de los interesados y Usuarios.

Fuente:[Propia]

➤ Alternativas y Competencias

Se ha mostrado interés en buscar alternativas externas para solucionar los diversos requerimientos, pero en la actualidad no existen herramientas en el mercado, puesto que al tratarse de una Universidad se necesita de un sistema personalizado.

2.1.6 Vista General del Producto

Esta sección provee información a alto nivel de las funciones del sistema implementado, partiendo de procesos principales como son: registro del currículum vitae de los investigadores, convocatorias a presentación de proyectos, registro de proyectos, evaluación de proyectos, aprobación de proyectos, oficialización de proyectos, ejecución de proyectos, cierre de proyectos, registro de eventos y registro de publicaciones de artículos científicos.

El sistema permite registrar investigadores y su currículum.

Permite crear una convocatoria de acuerdo a bases (política, área y línea de investigación) previamente definidas, ingresar evaluadores para una determinada convocatoria y la asignación de proyectos a los evaluadores.

Los proyectos que apliquen a la convocatoria realizada pasan por los procesos de presentación, evaluación, aprobación, oficialización, ejecución y cierre.

Con el sistema podemos crear convocatorias a diferentes eventos como: seminario, taller, capacitación, foro, video conferencia; los mismos que pueden ser generados como resultado a una investigación de algún proyecto.

También se puede registrar las publicaciones de artículos científicos realizadas por los investigadores.

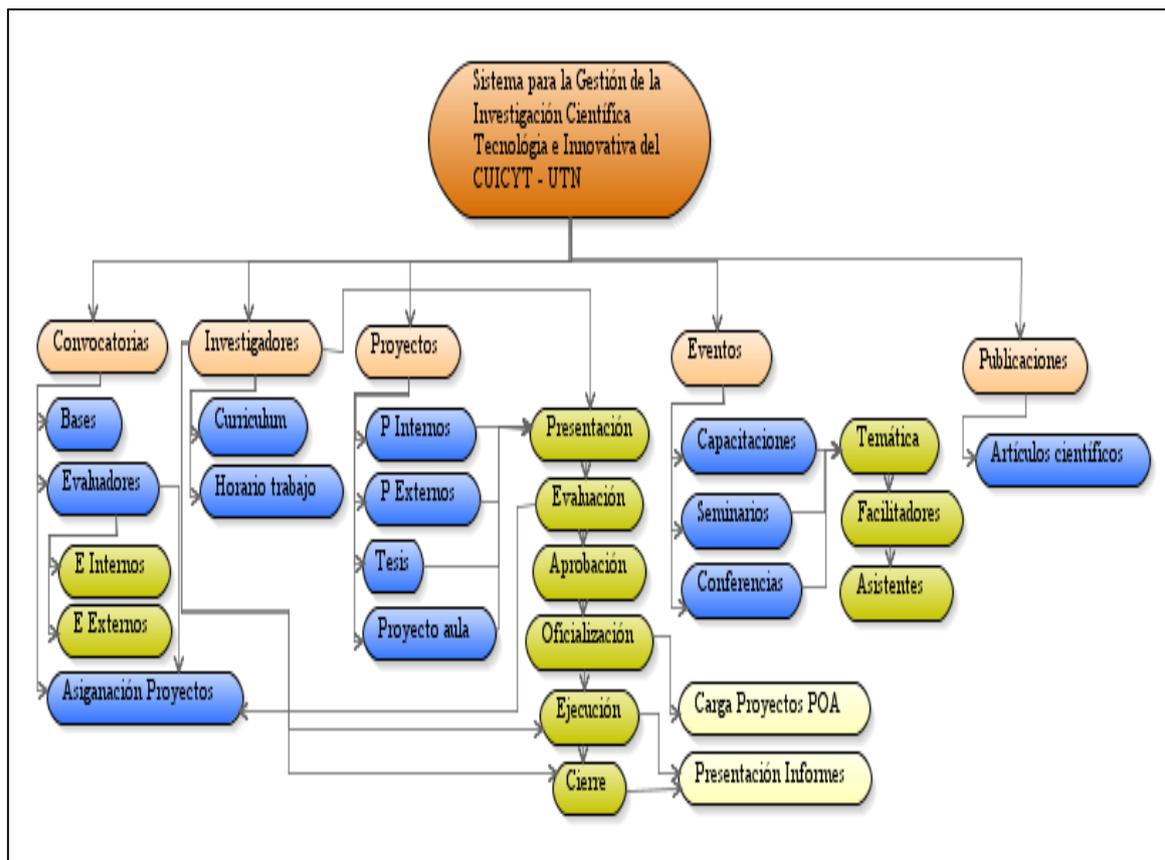


Figura 2.1. Perspectiva del Producto.
Fuente: [Propia].

➤ **Resumen de Capacidades**

Beneficios para el usuario	Características que lo soportan
El registro del escrutinio será completo y correcto automatizando reportes y estadística.	El ingreso de información será de forma sistematizada y ordenada.
Mejores tiempos de respuesta.	Al contar con un sistema computacional, los usuarios podrán ingresar simultáneamente los resultados de los escrutinios.
Se tendrá alta disponibilidad.	El acceso a la información a través de la Web permitirá a los usuarios un acceso inmediato desde cualquier punto de la intranet de la UTN.
Facilidades para el análisis de la información.	Brindará diferentes reportes y funciones de consulta.

Tabla 2.11. Resumen de Capacidades.
Fuente: [Propia].

➤ Suposiciones y Dependencias

Se asume que el CUICYT donde se encuentre establecido el sistema, tenga acceso TCP/IP al servidor de base de datos y de aplicaciones de la UTN. Esto con el objetivo de que cada usuario pueda acceder al producto, y le permita consultar y generar reportes de acuerdo a sus privilegios dentro del sistema.

➤ Costos y Precios

Detalle		USD	Real
Hardware	Equipos de Computación.	1'500.00	0.00
	Servidor de Aplicación Web.	5'000.00	0.00
	Servidor de Base de Datos.	5'000.00	0.00
	Equipo con Web Browser.	700.00	0.00
Software	Oracle® Standard One 10g (1 licencia procesador).	5'000.00	0.00
	Oracle® Developer Suite Release 10g	5'000.00	0.00
	Oracle® Application Server 10g,		
	Oracle® Forms, Report Server (1 licencia procesador).	20'000.00	0.00
Desarrollo	Pago por desarrollo.	1'000.00	0.00
Capacitación	Cursos y Libros, Asesoramiento.	800.00	0.00
Proyecto	Papelería y Suministros de Oficina.	500.00	0.00
Subtotal	(Parcial).	43'500.00	0.00
5% Imprevistos		240.00	0.00
Total		43'740.00	00.00

Tabla 2.12. Costos y Precios.

Fuente: [Propia].

➤ Licenciamiento e Instalación

La Universidad Técnica del Norte tiene licencia de desarrollador del producto Oracle® Developer Swite10g; con lo que el módulo puede ser diseñado libremente.

La instalación del producto es realizada por el personal de soporte del Departamento de Informática ya que es un sistema que utiliza tecnología Web.

2.1.7 Características del Producto

➤ Facilidad de Acceso y Uso

El “Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN”, será desarrollado en la herramienta Oracle Forms ® 10g, lo que permitirá a los usuarios un fácil acceso y uso.

➤ Unificación de la Información

Unos de los principales objetivos del “Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN”, es integrar y mantener una base de datos actualizada con todo lo referente a investigadores, convocatorias, proyectos, capacitaciones y publicaciones; para que esta información esté disponible en cualquier momento para el CUICYT y toda la comunidad universitaria a través de los reportes y búsquedas.

➤ Rangos de Calidad

El desarrollo del “Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN”, se elaborará siguiendo la Metodología de Desarrollo de Software RUP, contemplando los parámetros de calidad que la metodología define.

➤ Otros Requerimientos del Producto

Las Herramientas a utilizar para la elaboración del “Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN” son:

La interfaz de usuario estará diseñada en la herramienta Oracle ® Developer Suite versión 10g (Forms Developer - FormsBuilder).

El Servidor de Base de Datos será Oracle ® versión 10g.

Para consulta y análisis de información de la Base de Datos utilizaremos Oracle ® Discoverer versión 10g.

Para la elaboración de los Diagramas de los Artefactos de la metodología RUP se usará la herramienta JDeveloper ® versión 11.1.1.0.

Para la generación de reportes utilizaremos Oracle ® ReportsBuilder.

2.2 PLAN DE DESARROLLO DE SOFTWARE

Este plan de desarrollo de software es una versión preliminar preparada del "Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN", este documento provee una visión global del enfoque de desarrollo propuesto.

Para el proyecto utilizaremos metodología Rational Unified Process (RUP). Se incluirá el detalle para las fases de inicio y elaboración y adicionalmente se esbozarán las fases posteriores de construcción y transición para dar una visión global de todo el proceso.

El enfoque de desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

2.2.1 Propósito

El propósito del plan de desarrollo de software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del plan de desarrollo del software son:

- El jefe del proyecto, quien utiliza el presente plan para organizar la agenda y necesidades de recursos, y para realizar seguimiento.
- Los miembros del equipo de desarrollo, lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.

2.2.2 Alcance

El plan de desarrollo de software describe el plan global usado para el desarrollo del "Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN". El detalle de las iteraciones individuales se describe en los planes de cada iteración, documentos que se aportan en forma separada. Durante el proceso de desarrollo en el artefacto "Visión" se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones. Para la versión 0.1 del plan de desarrollo del software, nos hemos basado en la captura de requisitos por medio de entrevistas con el stakeholder, para hacer una estimación aproximada, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto "Visión", el cual se utilizará para refinar este documento. Posteriormente, el avance del

proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

2.2.3 Vista General del Proyecto

➤ Propósito, Alcance y Objetivos

La Universidad Técnica del Norte es una academia acreditada de carácter público, que contribuye al desarrollo integral de la sociedad. Forma profesionales emprendedores, competitivos, críticos, humanistas y éticos, comprometidos con el desarrollo sustentable. La Universidad Técnica del Norte en el año 2020, será un referente en ciencia, tecnología e innovación en el país, con estándares de excelencia internacionales. A través del CUICYT desea institucionalizar y sistematizar todo el proceso de investigación en ciencia, tecnología e innovación de manera interactiva y multidisciplinaria para propiciar la creación, adaptación, generación y transferencia de tecnología, en la búsqueda de alternativas viables e innovadoras para la solución de problemas prioritarios de los sectores sociales y productivos de la región norte del país.

El CUICYT a más de impulsar proyectos de investigación es un centro generador de investigadores a través de la investigación en el aula, formando grupos permanentes de estudiantes que se organizan en el aula alrededor de proyectos de investigación, con miras a proponer alternativas de solución a los problemas del entorno; realiza eventos tales como expo-ferias de investigación, talleres, seminarios publicación y difusión de los resultados de las investigaciones.

Esto conllevará una previsible adaptación a los nuevos sistemas de información y a la evolución tecnológica. Por ello, el CUICYT considera necesario el desarrollo de un "Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa" como parte del proceso de automatización de todas sus áreas institucionales.

➤ Suposiciones y Restricciones

Las suposiciones y restricciones respecto del "Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa del CUICYT-UTN", y que se derivan directamente de las entrevistas con el stakeholder del CUICYT son:

- a) El proyecto está completamente financiado por la Universidad Técnica del Norte y no habrá inconvenientes relacionados al costo total del proyecto ni a la agilidad con la que se deben atender los desembolsos parciales del mismo.
- b) Debe contemplarse las implicaciones de los siguientes puntos críticos:

- ✓ Gestión de flujos de trabajo, seguridad de transacciones e intercambio de información.
 - ✓ Adaptación al proceso académico de la UTN.
- c) Debe estar preparado para pruebas a realizarse en el año 2012.
- d) El sistema será diseñado sobre plataforma WEB y cumplirá con los estándares de calidad vigentes para desarrollo de software. Esto se conseguirá cumpliendo con el estándar PMI para dirección de proyectos, metodología RUP para el proceso de ingeniería de software y herramientas Oracle 10g para la construcción de las aplicaciones.

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

➤ **Entregables del Proyecto**

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto. Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración. A continuación se presenta la lista de artefactos propuesta para el proyecto:

- **Plan de Desarrollo del Software**

Es el presente documento.

- **Modelo de Casos de Uso del Negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

- **Visión**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

- **Glosario**

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología común.

- **Modelo de Casos de Uso**

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

- **Especificaciones de Casos de Uso**

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

- **Especificaciones Adicionales**

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.

- **Prototipos de Interfaces de Usuario**

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

- **Modelo de Análisis y Diseño**

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

- **Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

- **Modelo de Implementación**

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

- **Modelo de Despliegue**

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

- **Casos de Prueba**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

- **Solicitud de Cambio**

Los cambios propuestos para los artefactos se formalizan mediante este documento. Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación de impacto, y se asegura que éstos sean conocidos por el

equipo de desarrollo. Los cambios se establecen respecto de la última baseline (el estado del conjunto de los artefactos en un momento determinado del proyecto) establecida. En nuestro caso al final de cada iteración se establecerá una baseline.

- **Plan de Iteración**

Es un conjunto de actividades y tareas ordenadas temporalmente, con recursos asignados, dependencias entre ellas. Se realiza para cada iteración, y para todas las fases.

- **Evaluación de Iteración**

Este documento incluye la evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados.

- **Lista de Riesgos**

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

- **Manual de Instalación**

Este documento incluye las instrucciones para realizar la instalación del producto.

- **Material de Apoyo al Usuario Final**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento

- **Producto**

Los archivos del producto empaquetados y almacenados en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de construcción es desarrollado incrementalmente e interactivamente, obteniéndose una reléase al final de cada iteración.

Los artefactos 18, 19 y 20 se generarán a partir de la fase de Construcción, con lo cual se han incluido aquí sólo para dar una visión.

➤ **Evolución del Plan de Desarrollo del Software**

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

2.2.4 Organización del Proyecto

➤ Participantes en el Proyecto

El personal del proyecto, considerando las fases de Inicio, Elaboración y Construcción estará formado por los siguientes puestos de trabajo y personal asociado:

- **Jefe de Proyecto.** Labor asignada a Juan Carlos García, quien se encargará de organizar, planificar, coordinar y evaluar el desarrollo del proyecto.
- **Ingenieros de Software.** Perfiles asignados a Evelyn Enríquez y Luis Aguilar quienes realizarán labores de gestión de requisitos, gestión de configuración, documentación y diseño de datos. Encargados de las pruebas funcionales del sistema y labores de tester.
- **Analistas - Programadores.** Con conocimientos en el entorno de desarrollo del proyecto, con el fin de que los prototipos puedan ser lo más cercanos posibles al producto final, realizarán labores de gestión de requisitos, gestión de configuración, documentación y diseño de datos. Este trabajo ha sido encomendado Cristian Herrera.

➤ Interfaces Externas

La UTN definirá los participantes del proyecto que proporcionarán los requisitos del sistema, y entre ellos quiénes serán los encargados de evaluar los artefactos de acuerdo a cada subsistema y según el plan establecido.

El equipo de desarrollo interactuará activamente con los participantes de la UTN para especificación y validación de los artefactos generados.

➤ Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

Puesto	Responsabilidad
Jefe de Proyecto	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto.

	Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.
Analista de Sistemas	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.
Programador	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario
Ingeniero de Software	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.

Tabla 2.13. Roles y Responsabilidades.

Fuente: [Propia].

2.2.5 Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

Plan de las Fases.

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar)

Fase de Inicio	1	5 semanas
Fase de Elaboración	1	4 semanas
Fase de Construcción	1	18 semanas
Fase de Transición	1	2 semanas

Tabla 2.14. Plan de Fases.

Fuente: [Propia].

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de Inicio	En esta fase desarrollará los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente / usuario del artefacto

	Visión y el Plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de una semana.
Fase de Construcción	Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 0.3, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.
Fase de Transición	En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.

Tabla 2.15. Plan de Fases: Itos.

Fuente: [Propia].

➤ **Calendario del Proyecto**

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del

proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.

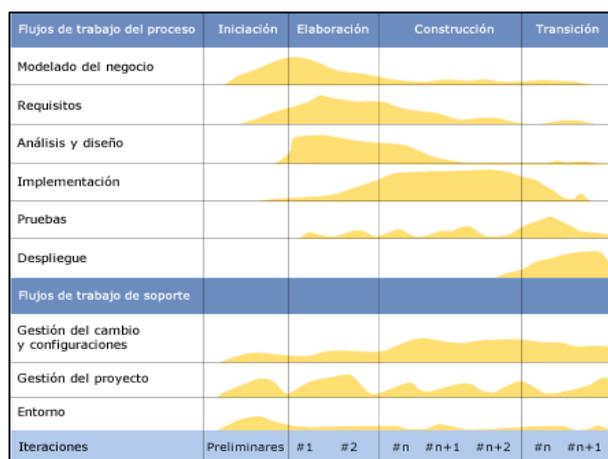


Figura 2.2. Descripción del Proceso Unificado.
Fuente: [W8].

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión está completo para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

Disciplinas / Artefactos Generados o Modificados Durante la Fase de Inicio	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 3 16/05 – 20/05	Semana 4 23/05 – 27/05
Requisitos		
Glosario	Semana 1 02/05 – 06/05	Semana 4 23/05 – 27/05
Visión	Semana 2 09/05 – 13/05	Semana 3 23/05 – 27/05
Modelo de Casos de Uso	Semana 2 09/05 – 13/05	siguiente fase
Especificación de Casos de Uso	Semana 4 23/05 – 27/05	siguiente fase
Especificaciones Adicionales	Semana 5	siguiente fase

[W8] WIKIPEDIA. (s.f.). Proceso Unificado de Rational , Recuperado el 14 de Diciembre del 2012, de http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational

	30/05 – 03/06	
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 6 06/06 – 10/06	siguiente fase
Modelo de Datos	Semana 7 13/06 – 17/06	siguiente fase
Implementación		
Prototipos de Interfaces de Usuario	Semana 8 20/06 – 24/06	siguiente fase
Modelo de Implementación	Semana 9 27/06 – 01/07	siguiente fase
Pruebas		
Casos de Pruebas Funcionales	Semana 10 04/07 – 08/07	siguiente fase
Despliegue		
Modelo de Despliegue	Semana 11 11/07 – 15/07	siguiente fase
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 0.1 y planes de las Iteraciones	Semana 1 02/05 – 06/05	Semana 2 09/05 – 13/05
Ambiente	Durante todo el proyecto	

Tabla 2.16. Calendario de la Fase de Inicio.

Fuente: [Propia].

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 3 16/05 – 20/05	Semana 4 23/05 – 27/05
Requisitos		
Glosario	Semana 2 09/05 – 13/05	Semana 4 23/05 – 27/05
Visión	Semana 1 09/05 – 13/05	Semana 3 23/05 – 27/05
Modelo de Casos de Uso	Semana 2 09/05 – 13/05	siguiente fase
Especificación de Casos de Uso	Semana 4 23/05 – 27/05	siguiente fase

Especificaciones Adicionales	Semana 5 30/05 – 03/06	siguiente fase
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 6 06/06 – 10/06	siguiente fase
Modelo de Datos	Semana 7 13/06 – 17/06	siguiente fase
Implementación		
Prototipos de Interfaces de Usuario	Semana 8 20/06 – 24/06	siguiente fase
Modelo de Implementación	Semana 9 27/06 – 01/07	siguiente fase
Pruebas		
Casos de Pruebas Funcionales	Semana 10 04/07 - 08/07	
Despliegue		
Modelo de despliegue	Semana 11	
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto	11/07 - -15/07	
Plan de Desarrollo del Software en su versión 0.2 y planes de las Iteraciones	Semana 10 04/07 – 08/07	siguiente fase
Ambiente	Durante todo el proyecto	

Tabla 2.17. Calendario de la Fase de Elaboración.

Fuente: [Propia].

2.2.6 Seguimiento y Control del Proyecto

➤ Gestión de Requisitos

Los requisitos del sistema son especificados en el artefacto Visión. Cada requisito tendrá atributos como importancia, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán gestionados mediante una Solicitud de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

- **Control de Plazos:** El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto y por el Comité de Seguimiento y Control.

- **Control de Calidad:** Los defectos encontrados en las revisiones y formalizados también en una Solicitud de Cambio tendrán un seguimiento para asegurar la conformidad en la solución de dichas deficiencias. Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist (listas de verificación) incluidas en RUP.
- **Gestión de Riesgos:** A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y las acciones establecidas como estrategia para mitigarlos o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.
- **Gestión de Configuración:** Se realizará gestión de configuración para llevar un registro de los artefactos generados y sus versiones. También se incluirá la Gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto. Al final de cada iteración se establecerá una base línea (un registro del estado de cada artefacto, estableciendo una versión), la cual podrá ser modificada sólo por una Solicitud de Cambio aprobada.

2.3 ESPECIFICACIÓN DE CASOS DE USO

2.3.1 Especificación Caso de Uso: Administrador del Sistema.

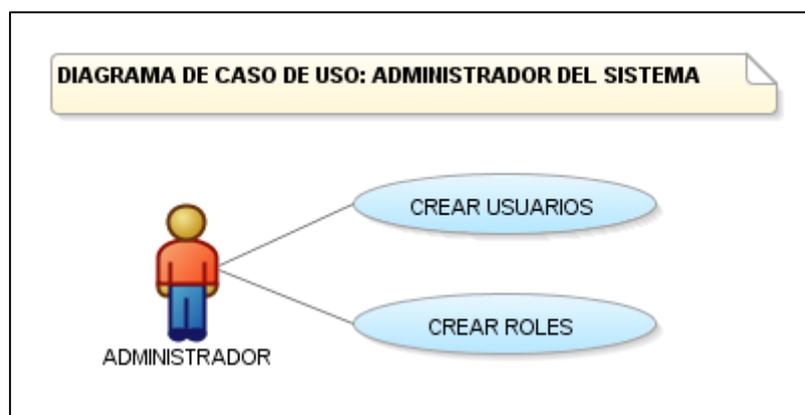


Figura 2.3. Caso de Uso Administrador del Sistema.
Fuente: [Propia].

➤ Descripción Breve

Este caso de uso describe los procesos que realiza el administrador del sistema para la creación de usuarios y roles.

➤ Flujo Básico de Eventos

- El administrador crea usuarios y Roles asignando los permisos necesarios en el Módulo de Seguridad.

Precondiciones

- ✓ El administrador del sistema se encuentre logeado y tenga en cuenta los permisos de usuario requeridos.
- ✓ El administrador del sistema debe conocer cuáles son las responsabilidades que tiene el usuario con el sistema.

Poscondiciones

- ✓ Dar los permisos de usuarios solicitados.
- ✓ Otorgar permisos en el menú del sistema de acuerdo al rol del usuario.

2.3.2 Especificación Caso de Uso: Investigadores

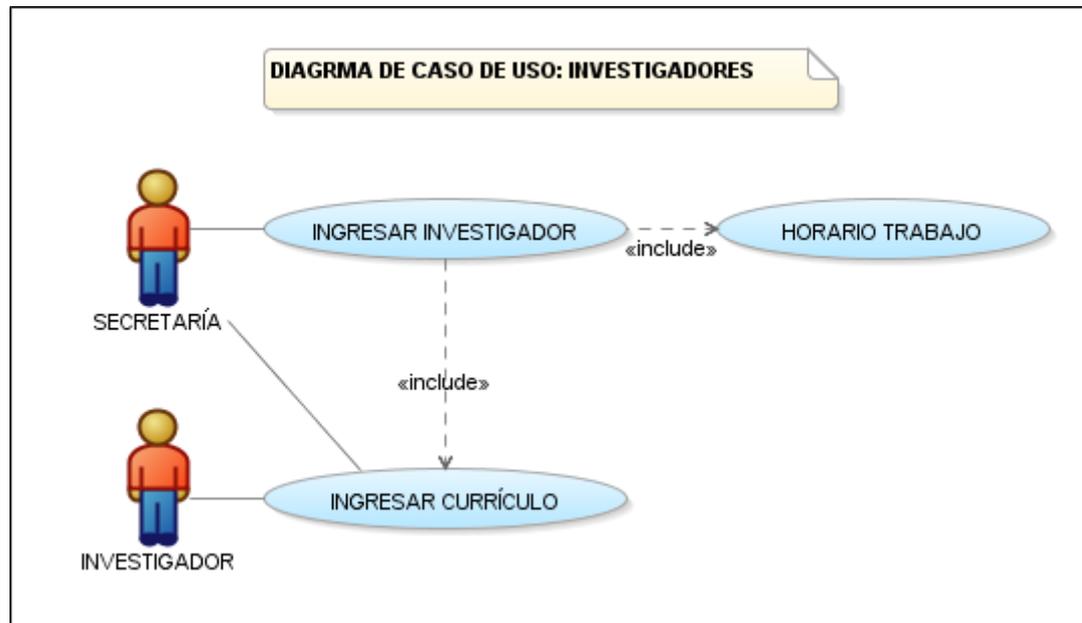


Figura 2.4. Caso de Uso Investigadores.
Fuente: [Propia].

➤ Descripción Breve

Este caso de uso describe todas las actividades que la secretaria o el investigador tiene que realizar para el registro del currículum del investigador, partiendo de los parámetros necesarios para el correcto desenvolvimiento del sistema hasta los procesos más complejos.

➤ Flujo Básico de Eventos

- La secretaria registra el currículum del investigador desde el formulario MGI_FRM_ADMINISTRAR_INVESTIGADORES.fmx.
- El investigador ingresa los datos de su currículum desde el formulario MGI_FRM_CURRICULUM_INVESTIGADORES.fmx.

Precondiciones

- ✓ Tanto la secretaria como el investigador deben tener los permisos pertinentes en el sistema para poder ingresar la información requerida.

Poscondiciones

- ✓ El perfil del investigador queda registrado.

2.3.3 Especificación Caso de Uso: Convocatorias

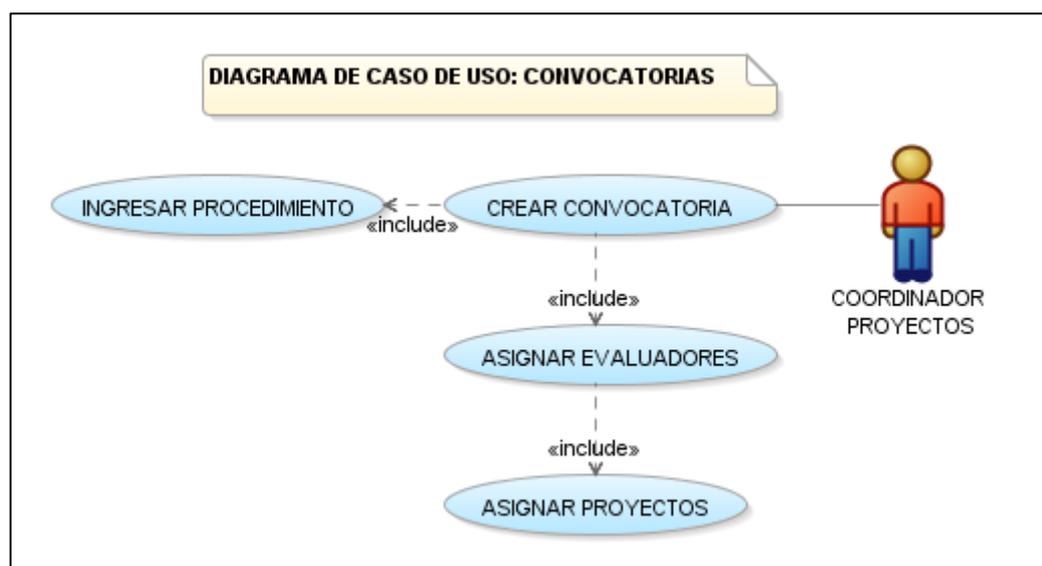


Figura 2.5. Caso de Uso Convocatorias.
Fuente: [Propia].

➤ Descripción Breve

Este caso de uso describe todas las actividades que el coordinador de proyectos puede realizar para realizar convocatorias, partiendo de los parámetros necesarios para el correcto desenvolvimiento del sistema hasta los procesos más complejos.

➤ Flujo Básico de Eventos

- El coordinador de proyectos puede crear convocatorias a presentación de proyectos o convocatorias a eventos a través de lineamientos como son las políticas, áreas y líneas de investigación. Forma MGI_FRM_CONVOCATORIA.fmx.

Precondiciones

- ✓ El usuario debe tener los permisos necesarios en el sistema.
- ✓ La convocatoria debe especificar el formato al cual los proyectos tienen que registrarse, para eso se utiliza la forma MGI_FRM_FORMATO_PROYECTO.fmx
- ✓ Las políticas de investigación deben ser definidas a través de la forma MGI_FRM_POLITICAS_INVESTIGACION.fmx.
- ✓ Las áreas de investigación deben ser definidas a través de la forma MGI_FRM_AREAS_INVESTIGACION.fmx.
- ✓ Las líneas de investigación deben ser definidas a través de la forma

MGI_FRM_LINEAS_INVESTIGACION.fmx.

Poscondiciones

- ✓ Ingresar el procedimiento a seguir para la presentación de proyectos.
 - ✓ Asignar evaluadores a la convocatoria.
 - ✓ Asignar proyectos a los evaluadores.
 - ✓ Evaluar proyectos.
- Ingresar el procedimiento a cumplir por parte de los que presenten proyectos. Forma: MGI_FRM_PROCEDIMIENTO_CONV.fmx.

Precondiciones

- ✓ La convocatoria debe estar creada.
 - ✓ El usuario debe tener privilegios para acceder a esta instancia.
- Asignar evaluadores. Forma: MGI_FRM_EVALUADORES.fmx.

Precondiciones

- ✓ Permisos de usuario en el sistema.
- ✓ Debe existir una convocatoria.
- ✓ Deben estar registrado el evaluador en el sistema ya sea como un investigador, como docente o como una persona externa a la universidad.

Poscondiciones

- ✓ Se puede asignar proyectos al evaluador de acuerdo al área de investigación en el que se encuentre el proyecto.
- ✓ Se puede evaluar proyectos.

2.3.4 Especificación Caso de Uso: Presentación de Proyectos

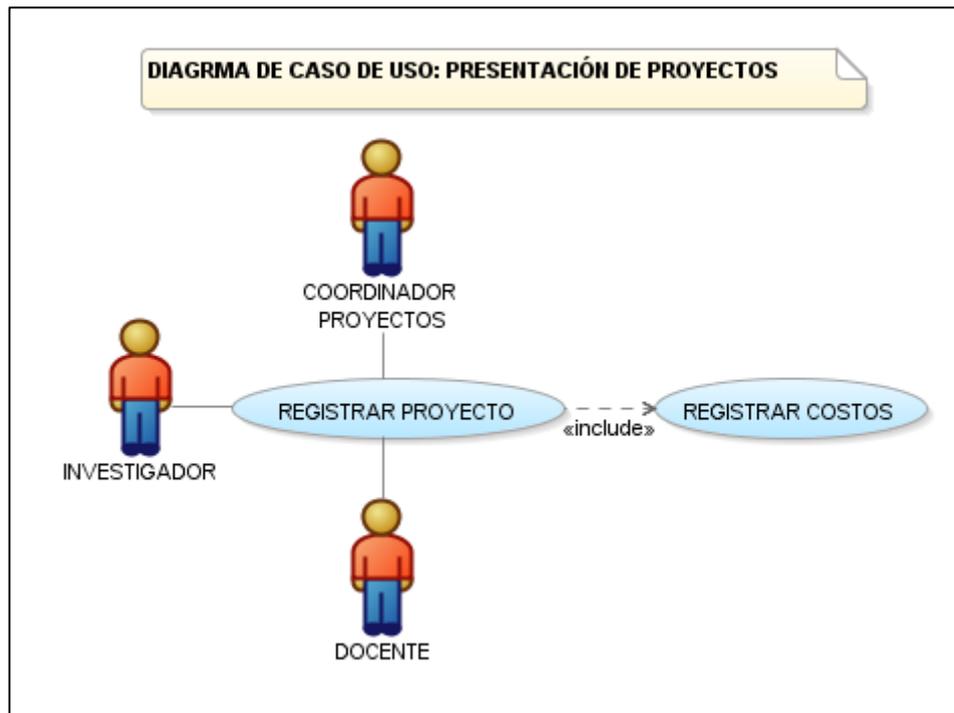


Figura 2.6. Caso de Uso Presentación de Proyectos.
Fuente: [Propia].

➤ Descripción Breve

A través de este caso de uso el usuario puede registrar su proyecto, aplicando a una convocatoria.

➤ Flujo Básico de Eventos

- El usuario registra el proyecto, ingresando los datos técnicos a través de la forma MGI_FRM_PROYECTOS.fmx. e ingresando el costo por medio de la forma MGI_FRM_COSTOS_PROYECTO.fmx

Precondiciones

- ✓ Debe existir una convocatoria realizada.
- ✓ Bebe haber investigadores, docentes, o estudiantes registrados.

Poscondiciones

- ✓ Se puede ingresar los costos que demanda el proyecto.
- ✓ Puede ser asignado a un evaluador para su posterior revisión.

2.3.5 Especificación Caso de Uso: Evaluación de Proyectos

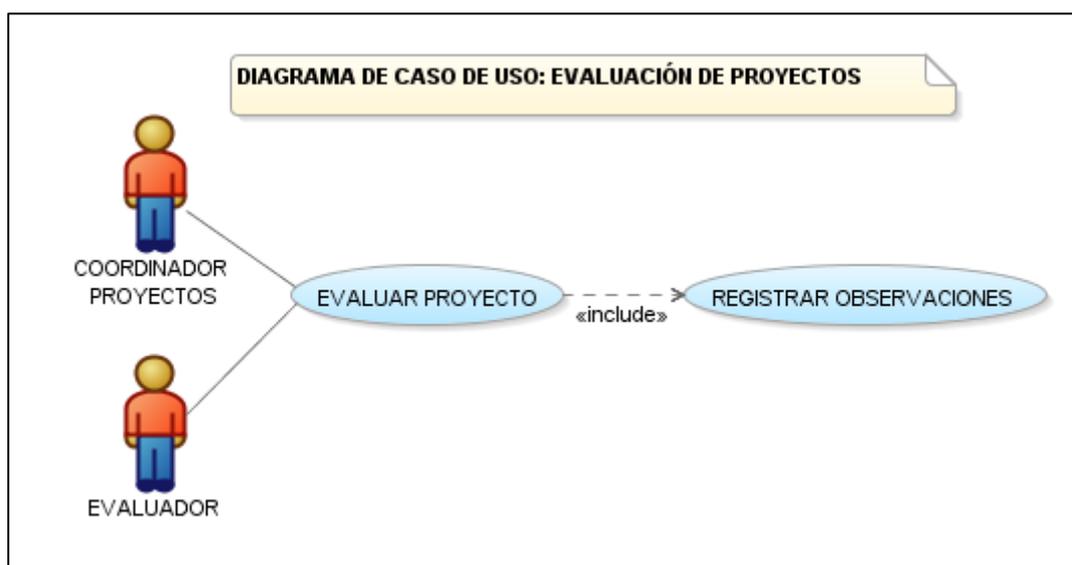


Figura 2.7. Caso de Uso Evaluación de Proyectos.
Fuente: [Propia].

➤ Descripción Breve

Por medio de esta caso de uso el coordinador de proyectos o el evaluador puede evaluar el proyecto, partiendo de los parámetros necesarios para el correcto desenvolvimiento del sistema hasta los procesos más complejos.

➤ Flujo Básico de Eventos

- Evaluar Proyecto. Forma: MGI_FRM_EVALUAR_PROYECTO.fmx.

Precondiciones

- ✓ Debe existir una convocatoria realizada.
- ✓ Bebe haber investigadores, docentes, o estudiantes registrados.
- ✓ Los parámetros de evaluación deben estar establecidos a través de la forma MGI_FRM_PARAMETROS_EVALUACION.fmx.
- ✓ Debe haber proyectos asignados a un evaluador.

Poscondiciones

- ✓ Se puede ingresar los costos que demanda el proyecto.
- ✓ El proyecto puede ser asignado a un evaluador para su posterior revisión.

- ✓ Se puede registrar las observaciones realizadas al proyecto.
- ✓ Después que el proyecto es evaluado, de acuerdo al puntaje obtenido este puede ser aprobado.
- Registrar Observaciones. Forma: MGI_FRM_OBSERVACIONES.fmx.

Precondiciones

- ✓ El proyecto debe estar evaluado con uno o todos los parámetros de evaluación.

Poscondiciones

- ✓ Las observaciones realizadas al proyecto pueden ser tomadas en cuenta.

2.3.6 Especificación Caso de Uso: Aprobación de Proyectos

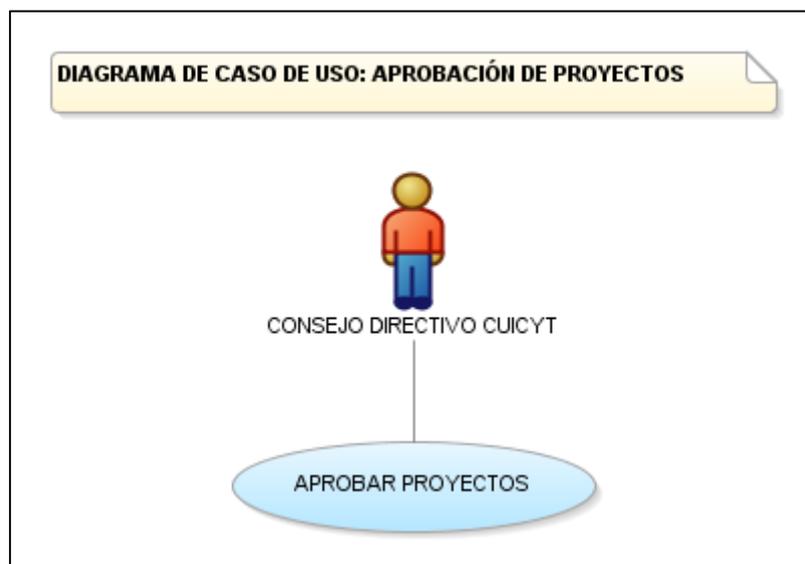


Figura 2.8. Caso de Uso Aprobación de Proyectos.

Fuente: [Propia].

➤ Descripción Breve

Este caso de uso describe el proceso que realiza el usuario del sistema para aprobar los proyectos.

➤ Flujo Básico de Eventos

- Aprobar Proyectos. Forma: MGI_FRM_APROBAR_PROYECTOS.fmx.

Precondiciones

- ✓ El proyecto debe estar evaluado por los responsables pertinentes.

- ✓ Para ser aprobado el proyecto debe tener como mínimo el porcentaje establecido en la convocatoria realizada.

Poscondiciones

- ✓ Una vez aprobado el proyecto puede ser considerado a formar parte del grupo de proyectos a cargarse en el Módulo de Planeamiento Estratégico.

2.3.7 Especificación Caso de Uso: Oficialización de Proyectos

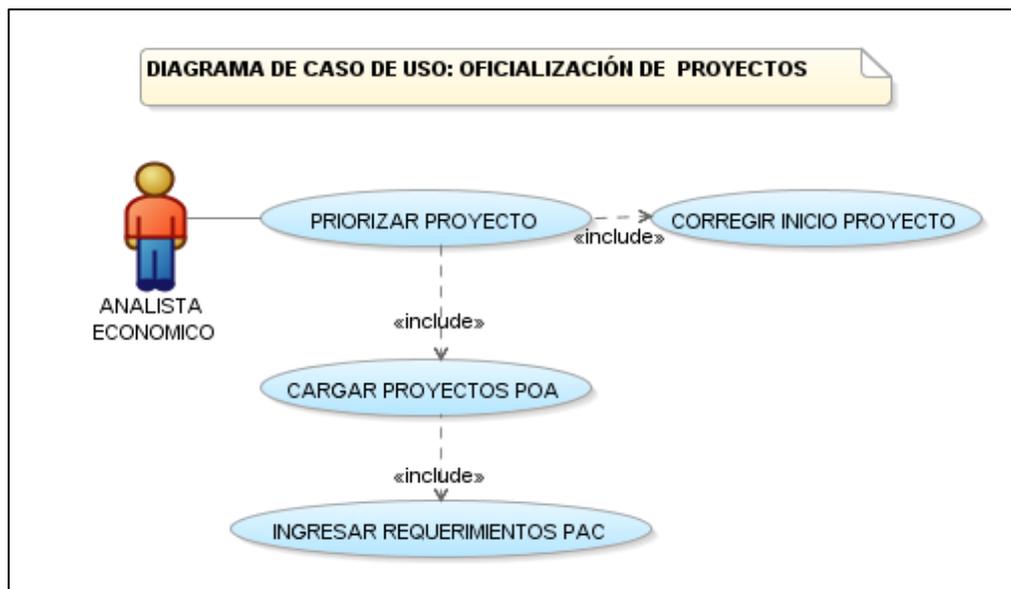


Figura 2.9. Caso de Uso Oficialización de Proyectos.
Fuente: [Propia].

➤ Descripción Breve

Con este caso de uso se prioriza los proyectos aprobados, se corrige la fecha de inicio y de finalización, se carga al Módulo de Planeamiento Estratégico y finalmente se ingresa los requerimientos que demandan al Plan Anual de Compras (PAC).

➤ Flujo Básico de Eventos

- Priorizar Proyectos. Forma: MGI_FRM_PRIORIZAR_PROYECTOS.fmx.

Precondiciones

- ✓ El proyecto debe estar aprobado.

Poscondiciones

- ✓ Se puede comenzar la ejecución del proyecto.

2.3.8 Especificación Caso de Uso: Ejecución de Proyectos

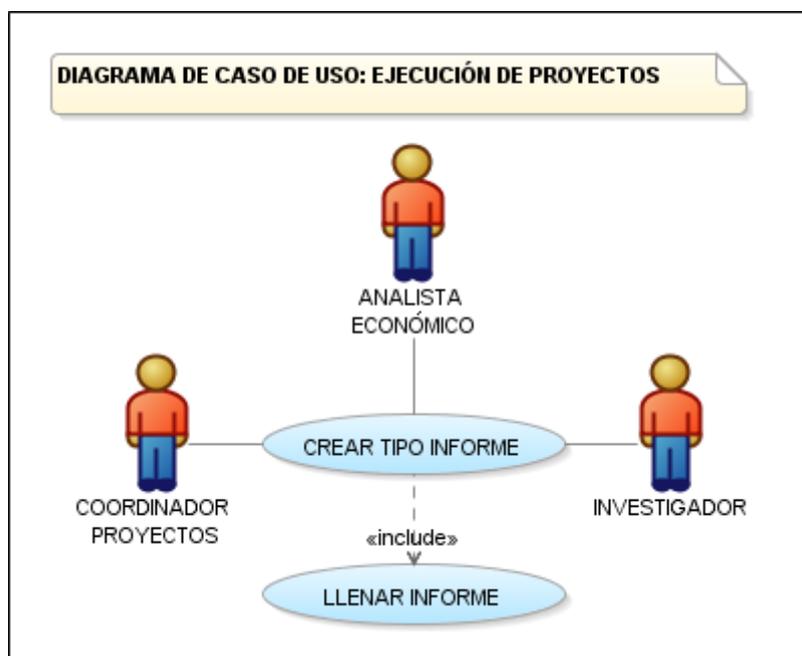


Figura 2.10. Caso de Uso Ejecución de Proyectos.
Fuente: [Propia].

➤ Descripción Breve.

Este caso de uso describe el proceso que realiza el investigador para dar inicio a la ejecución del proyecto, a través de la presentación de informes.

➤ Flujo Básico de Eventos

- Crear Informe. Forma: MGI_FRM_INFORME_PROYECTO.fmx.

Precondiciones

- ✓ El proyecto debe estar cargado al Módulo de Planeamiento Estratégico.
- ✓ Los costos con sus actividades respectivas deben constar en el Módulo de Planeamiento Estratégico.

Poscondiciones

- ✓ El proyecto puede avanzar en su desarrollo a través de la presentación de informes.

2.3.9 Especificación Caso de Uso: Cierre de Proyectos

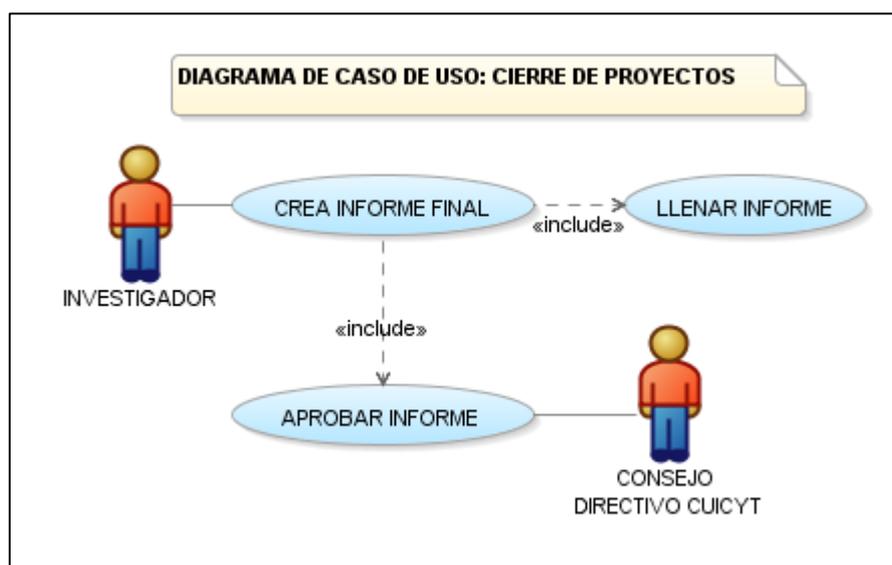


Figura 2.11. Caso de Uso Cierre de Proyectos.
Fuente: [Propia].

➤ Descripción Breve

Este caso de uso describe las actividades necesarias que debe realizar el usuario para dar por terminado la ejecución del proyecto.

➤ Flujo Básico de Eventos

- Crear Informe Final.

Precondiciones

- ✓ El coordinador de proyectos ha realizado correctamente el login en el sistema.
- ✓ El coordinador de proyectos ha seleccionado el botón de “Formatos Proyectos” de su interfaz.

Poscondiciones

- ✓ En caso de haberse dado de alta un estado de proyecto, los datos del mismo quedan almacenados en la base de datos.

2.3.10 Especificación Caso de Uso: Eventos

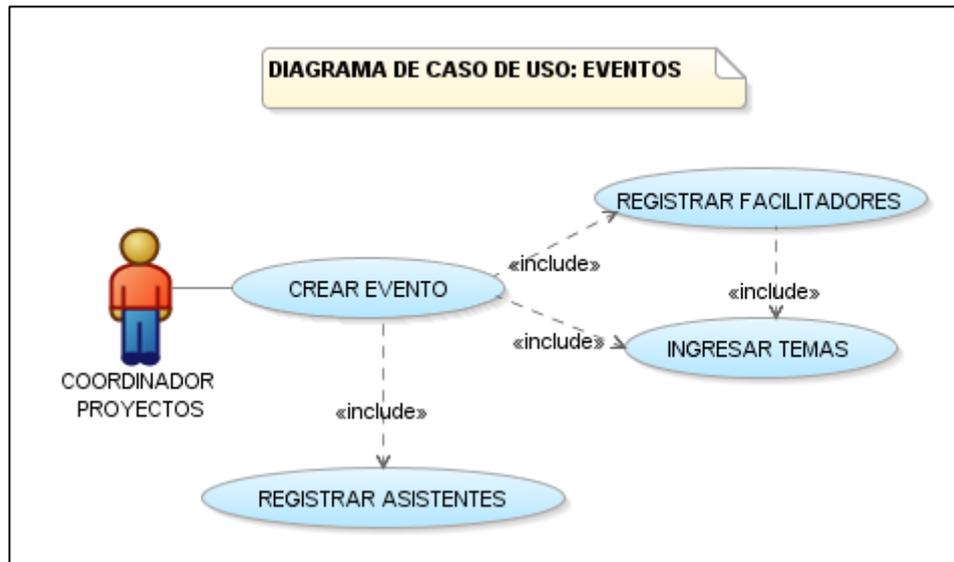


Figura 2.12. Caso de Uso Eventos.
Fuente: [Propia].

➤ Descripción Breve

El caso de uso permite al usuario crear diferentes eventos, los mismos que pueden ser generados como producto de una investigación.

➤ Flujo Básico de Eventos

- Crear Evento. Forma: MGI_FRM_EVENTOS.fmx

Precondiciones

- ✓ Debe existir tipos de eventos definidos.
- ✓ Debe estar creada una convocatoria a un evento.

Poscondiciones

- ✓ Se puede registrar los facilitadores, los temas presentados por parte del facilitador, los asistentes al evento.
- Registrar Facilitadores. Forma: MGI_FRM_FACILITADORES.fmx

Precondiciones

- ✓ Tener en cuenta el tipo de facilitador.

- ✓ El evento tiene que estar creado.

Poscondiciones

- ✓ Se puede registrar el tema a ser presentado por el facilitador.
- Ingresar Temas. Forma: MGI_FRM_CONTENIDO_EVENTO.fmx.

Precondiciones

- ✓ Deben estar registrados los facilitadores del evento.

Poscondiciones

- ✓ Se puede insertar varios temas para un mismo facilitador.

2.3.11 Especificación Caso de Uso: Publicaciones

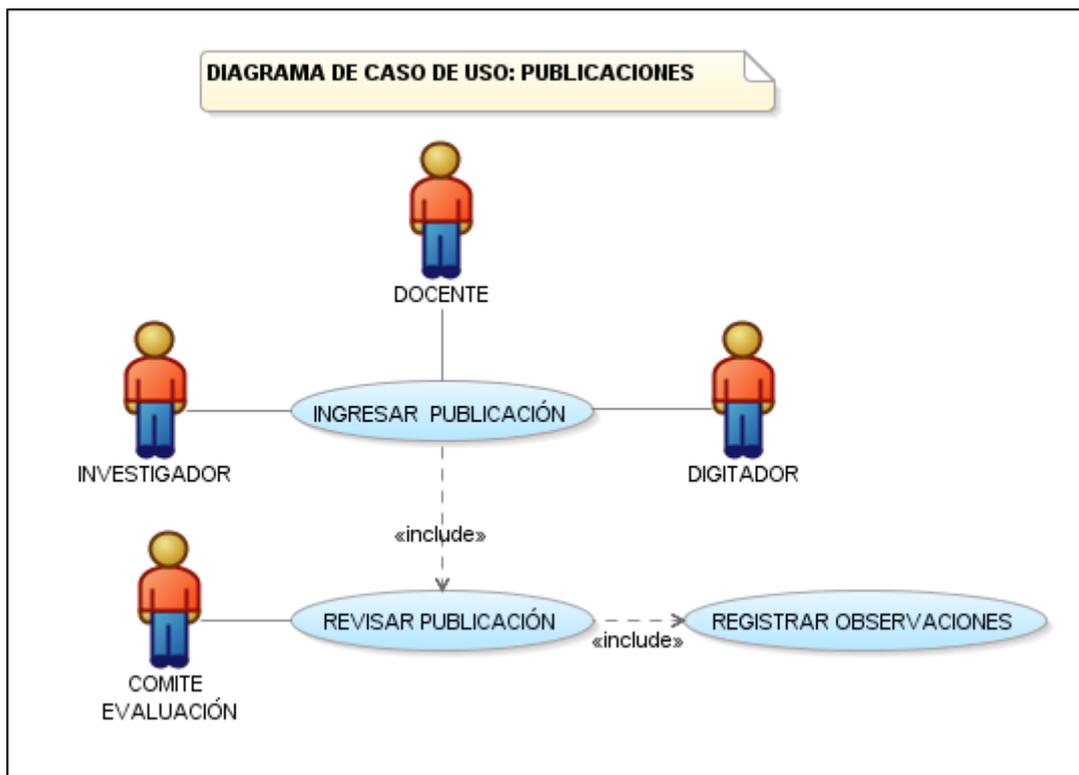


Figura 2.13. Caso de Uso Publicaciones.

Fuente: [Propia].

➤ Descripción Breve

Este caso de uso describe los procesos que realiza el administrador del sistema para la creación roles, apertura y finalización de convocatorias para la recepción de proyectos.

➤ **Flujo Básico de Eventos**

- El administrador crea usuarios y Roles asignando los permisos necesarios en el Módulo de Seguridad.
- Crea convocatorias de acuerdo a los lineamientos del CUICYT, en el formulario

Precondiciones

- ✓ El coordinador de proyectos ha realizado correctamente el login en el sistema.
- ✓ El coordinador de proyectos ha seleccionado el botón de “Formatos Proyectos” de su interfaz.

Poscondiciones

- ✓ En caso de haberse dado de alta un estado de proyecto, los datos del mismo quedan almacenados en la base de datos.

2.4 VISTA LÓGICA

2.4.1 Modelo Entidad Relación

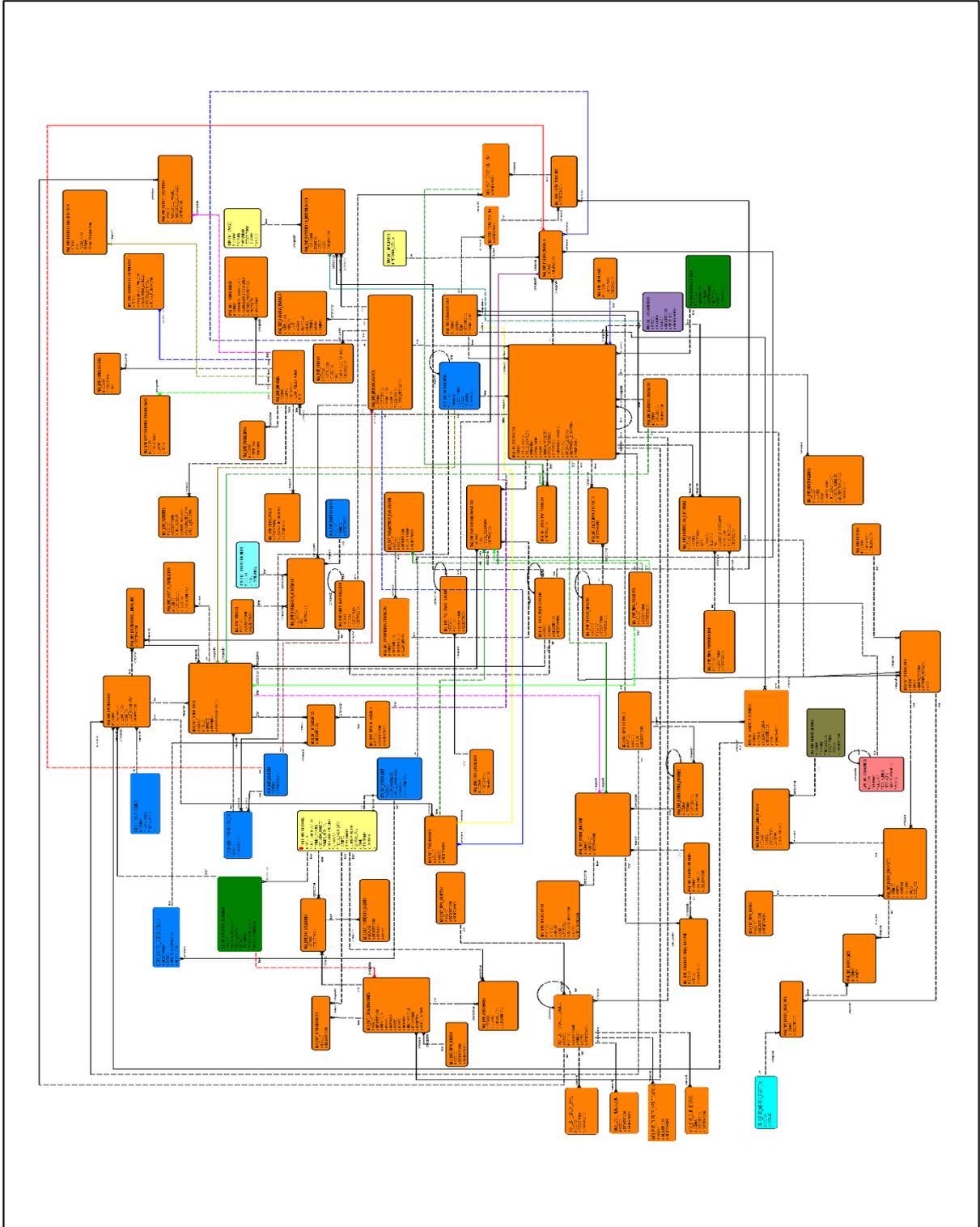


Figura 2.14. Modelo Entidad Relación.
Fuente: [Propia].

2.5 VISTA DE IMPLEMENTACIÓN

2.5.1 Diagramas de Actividades

➤ Crear Usuarios

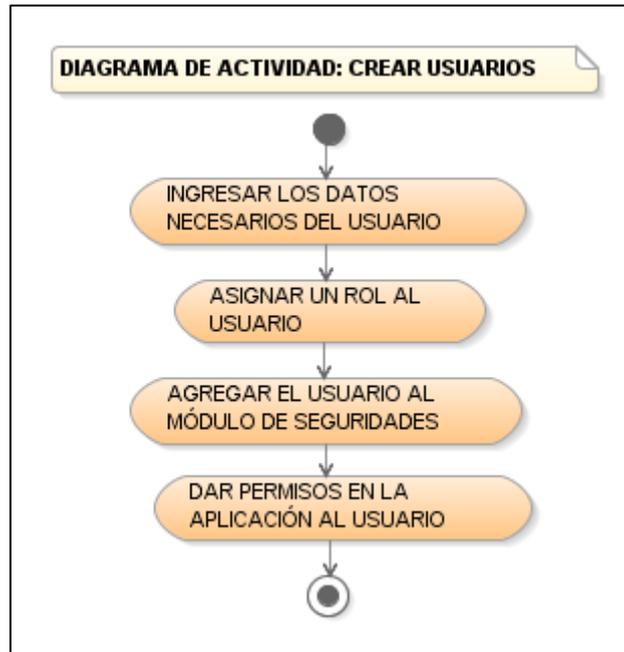


Figura 2.15. Diagrama de Actividad Crear Usuarios.
Fuente: [Propia].

➤ Crear Roles

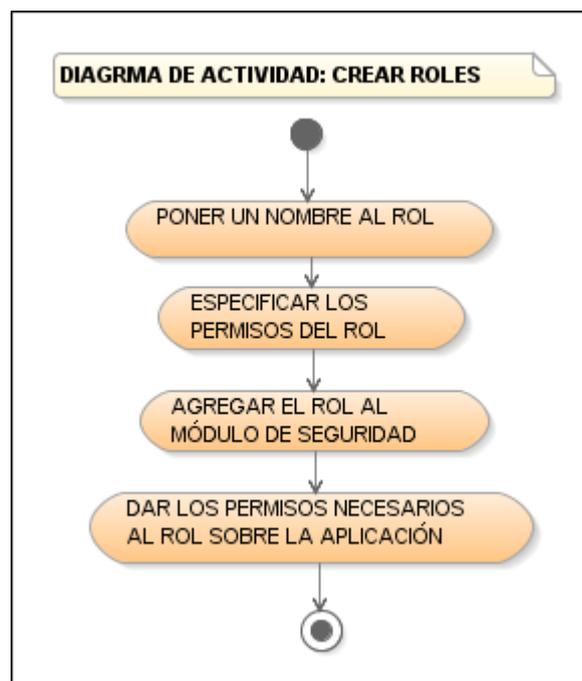


Figura 2.16. Diagrama de Actividad Crear Roles.
Fuente: [Propia].

➤ **Ingresar Investigador**

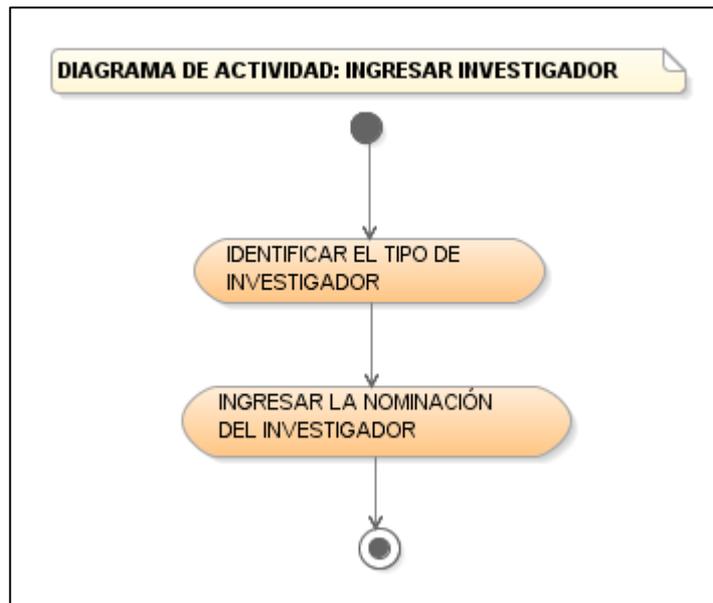


Figura 2.17. Diagrama de Actividad Ingresar Investigador.
Fuente: [Propia].

➤ **Ingresar Horario de Trabajo**



Figura 2.18. Diagrama de Actividad Ingresar Horario de Trabajo.
Fuente: [Propia].

➤ **Ingresar Currículo**

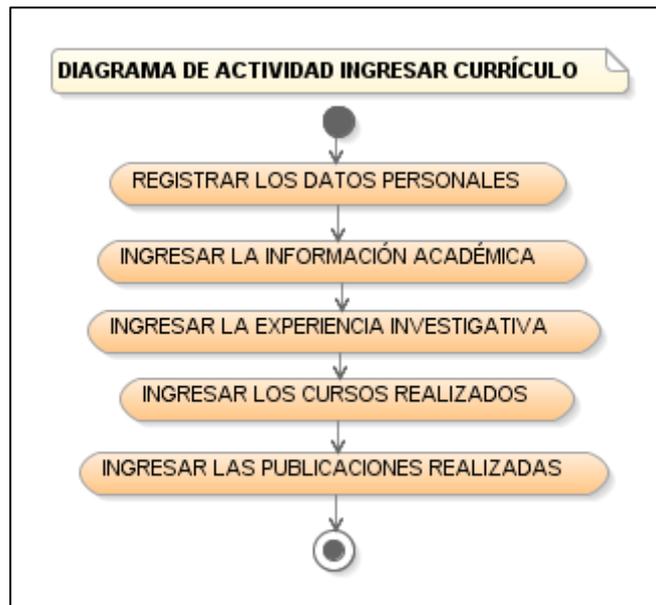


Figura 2.19. Diagrama de Actividad Ingresar Currículo.
Fuente: [Propia].

➤ **Crear Convocatoria**

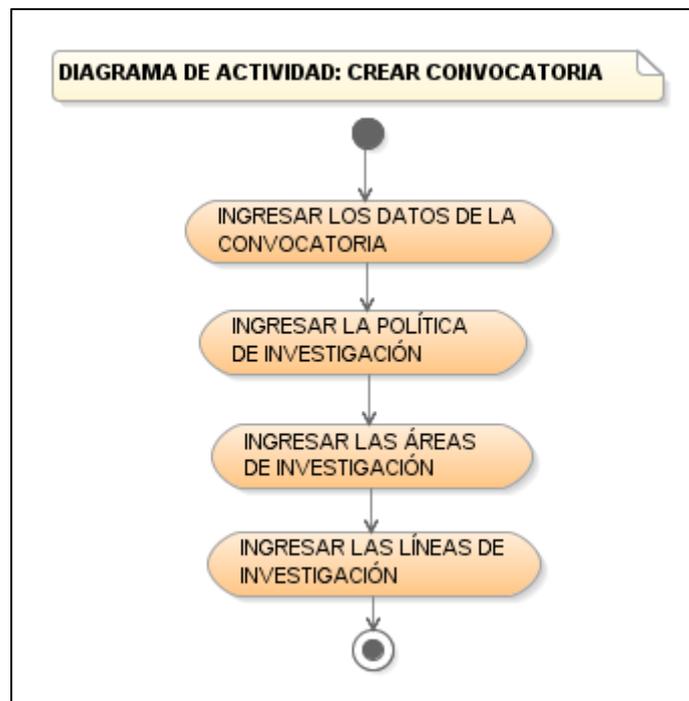


Figura 2.20. Diagrama de Actividad Crear Convocatoria.
Fuente: [Propia].

➤ **Ingresar Procedimiento**

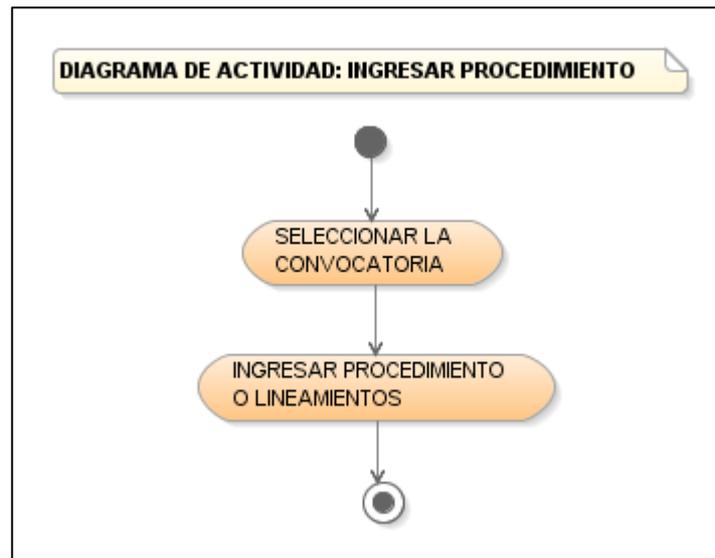


Figura 2.21. Diagrama de Actividad Ingresar Procedimiento.
Fuente: [Propia].

➤ **Asignar Evaluadores**

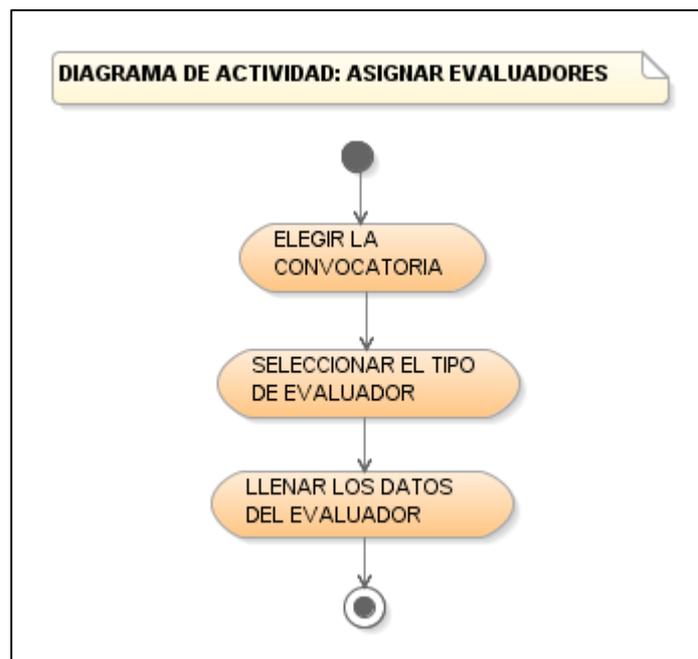


Figura 2.22. Diagrama de Actividad Asignar Evaluadores.
Fuente: [Propia].

➤ **Asignar Proyectos**

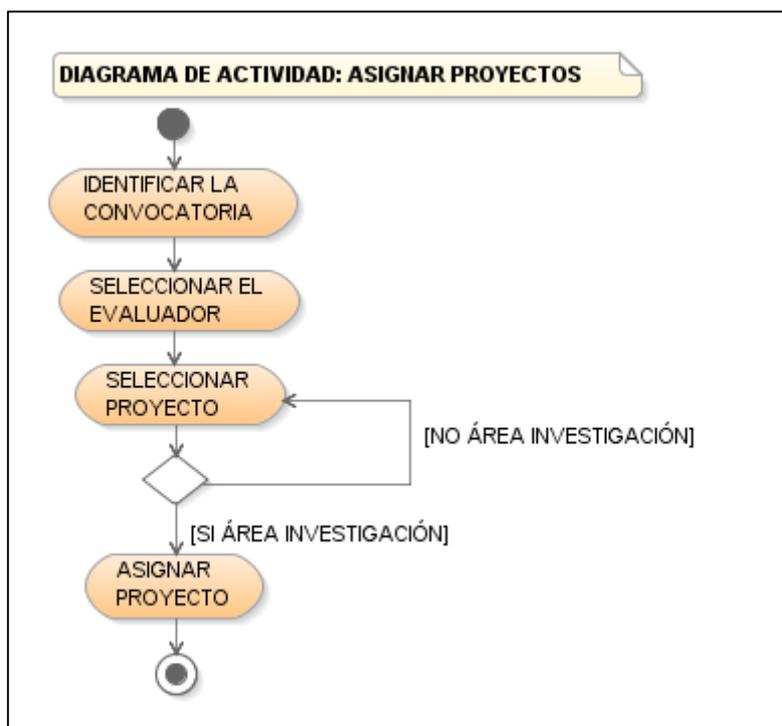


Figura 2.23. Diagrama de Actividad Asignar Proyectos.
Fuente:[Propia].

➤ **Registrar Proyecto**

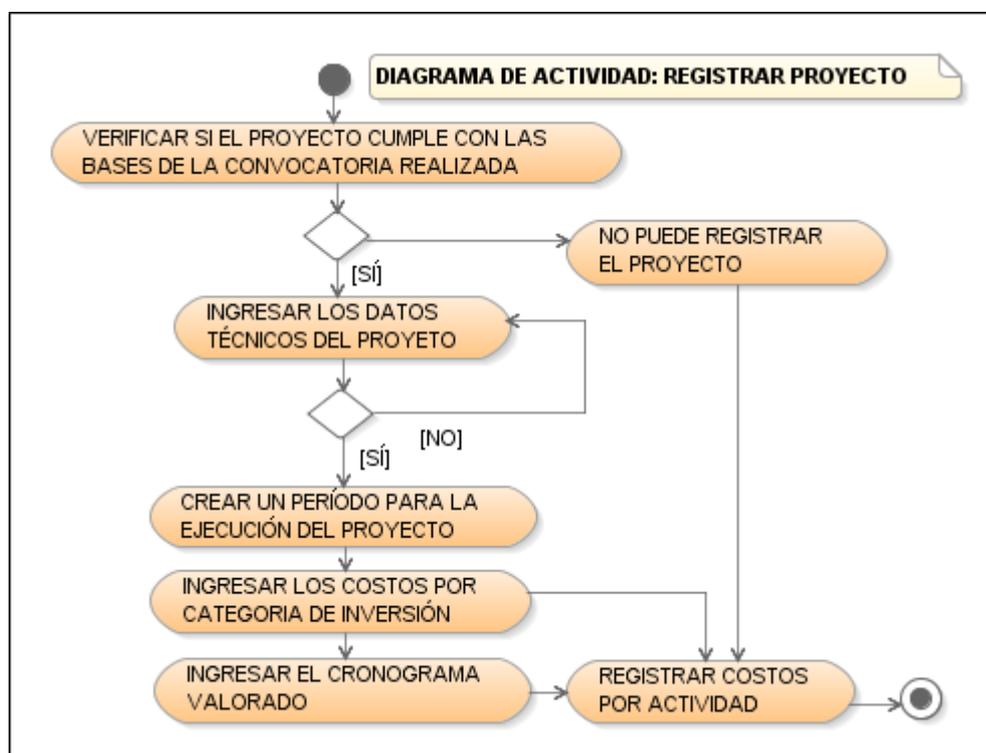


Figura 2.24. Diagrama de Actividad Registrar Proyecto.
Fuente: [Propia].

➤ **Evaluar Proyecto**

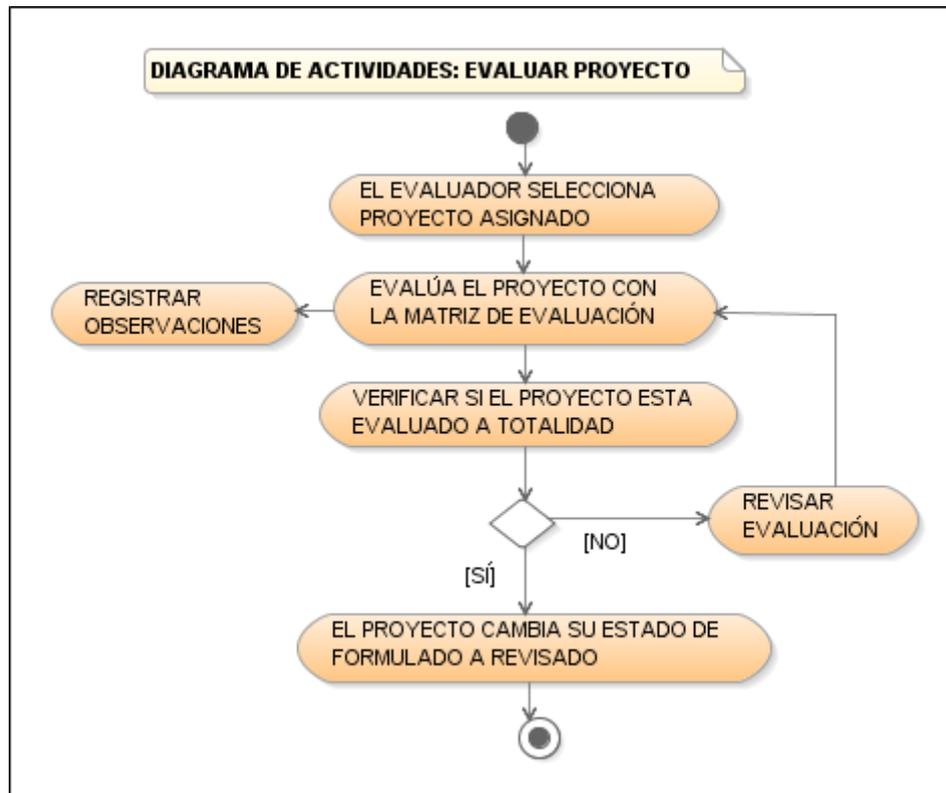


Figura 2.25. Diagrama de Actividad Evaluar Proyecto.
Fuente: [Propia].

➤ **Aprobar Proyecto**

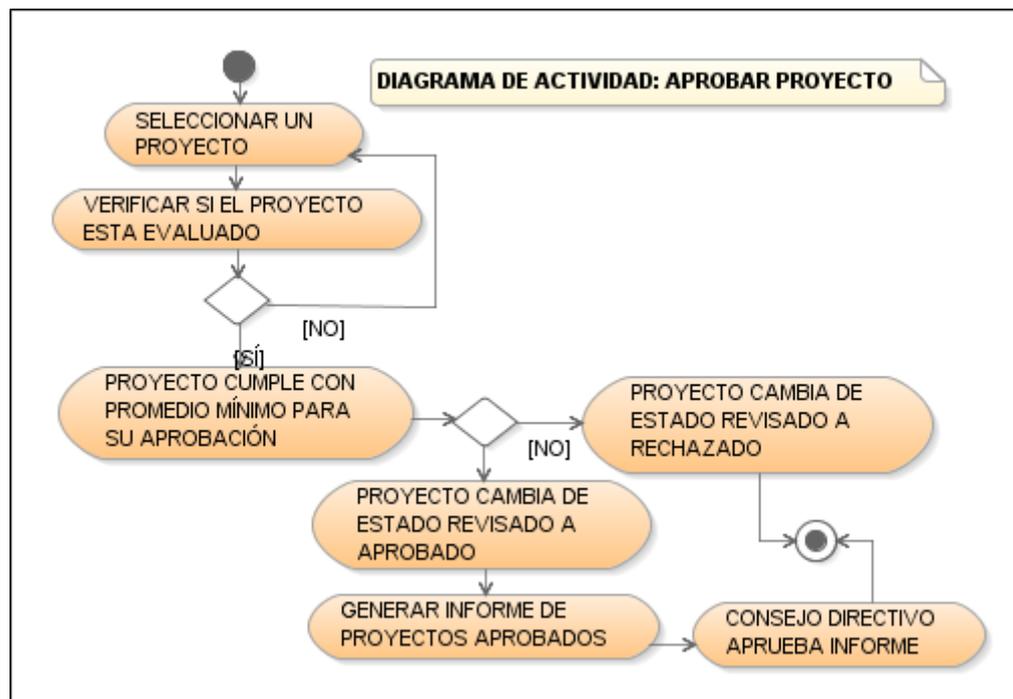


Figura 2.26. Diagrama de Actividad Aprobar Proyecto.
Fuente: [Propia].

➤ **Oficializar Proyecto**

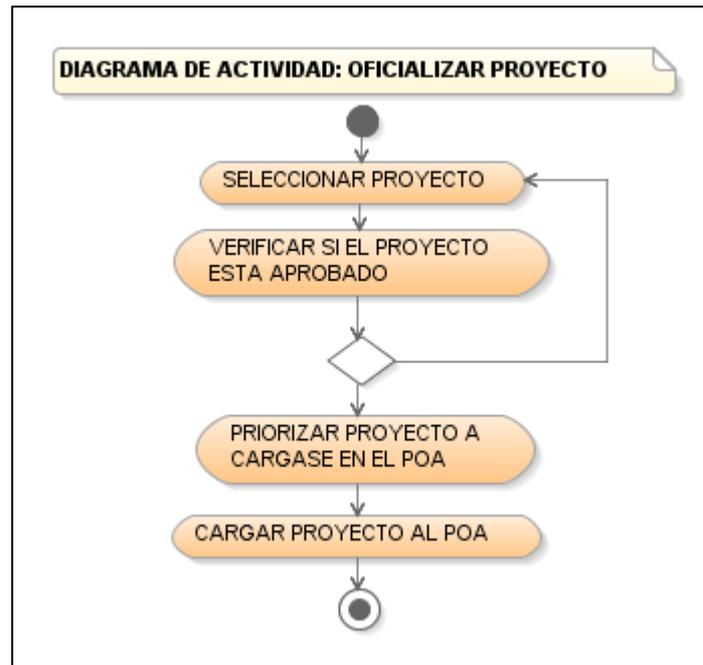


Figura 2.27. Diagrama de Actividad Oficializar Proyecto.
Fuente: [Propia].

➤ **Ejecutar y Cerrar Proyecto**

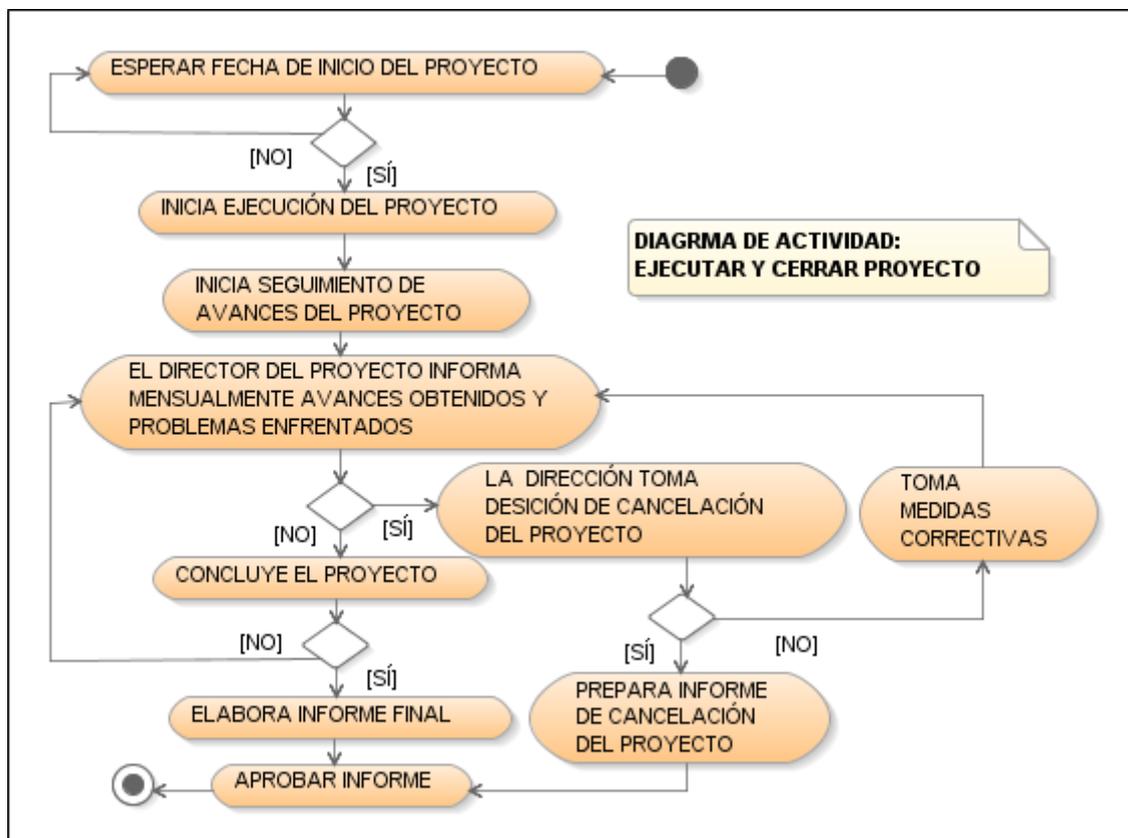


Figura 2.28. Diagrama de Actividad Ejecutar y Cerrar Proyecto.
Fuente: [Propia].

2.5.2 Diagramas de Componentes

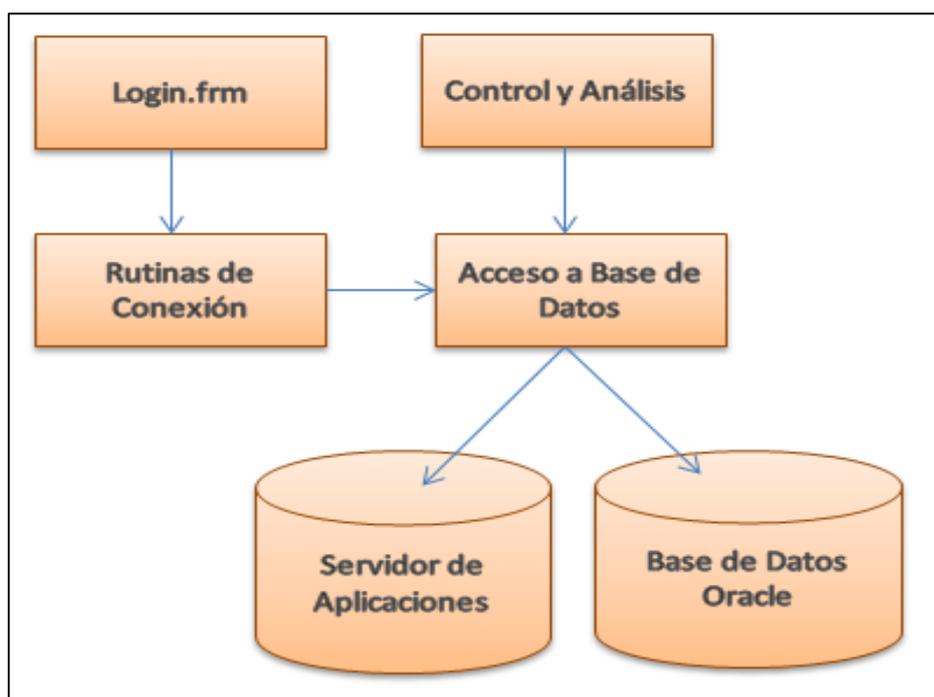


Figura 2.29. Diagrama de Componentes.
Fuente: [Propia].

2.5.3 Diagrama de Arquitectura

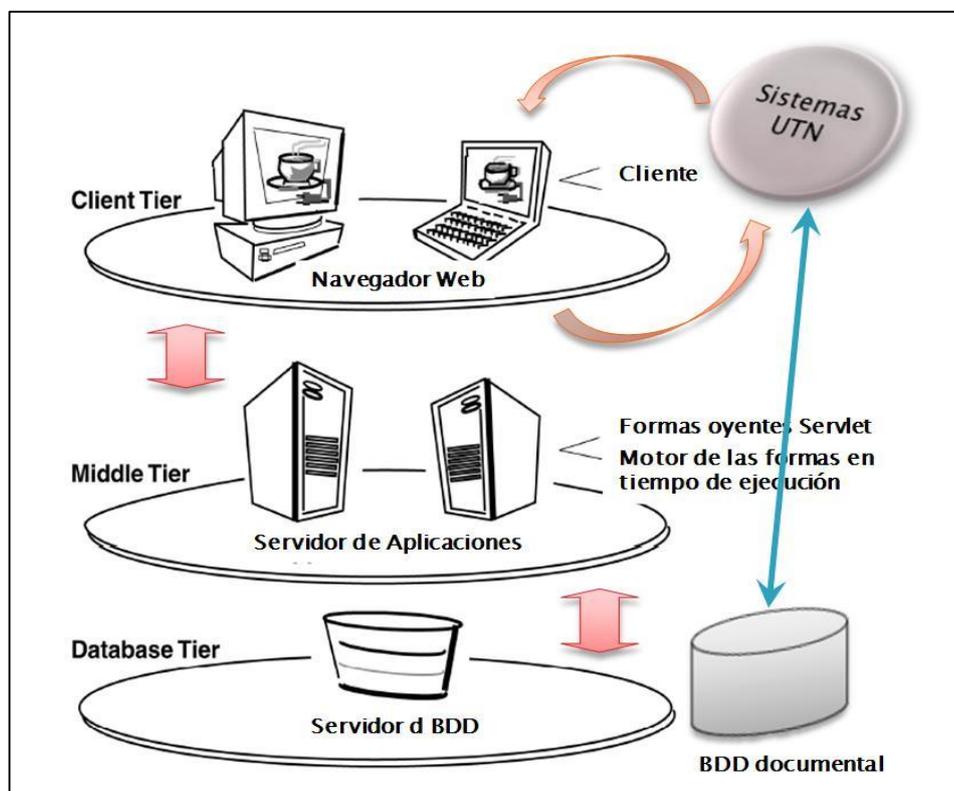


Figura 2.30. Diagrama de Arquitectura.
Fuente:[Propia].

2.6 CASOS DE PRUEBA

Para la elaboración de las pruebas se creó un usuario “mgi_pruebas”, se le proporcionó los permisos necesarios para poder acceder al sistema y a sus funcionalidades.

2.6.1 Especificación de Caso de Prueba: Ingresar Investigador

➤ Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Investigadores: Ingresar Investigador”. La única prueba que se puede comprobar es que el usuario ingrese bien la información. El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ Comprobar la Manipulación de Datos

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ Condiciones de Ejecución

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ Entrada

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: mantenimiento, investigadores y damos clic en ingresar investigador.
- Llenar los datos relevantes del documento y guardar;

➤ Resultado Esperado

- Que el usuario pueda ingresar correctamente la información.

➤ Evaluación de la Prueba

- Prueba superada con éxito.

- Para mayor funcionalidad se agregó un botón con el que el usuario pueda desplegar fácilmente la lista de valores en el campo cédula.

2.6.2 Especificación de Caso de Prueba: Ingresar Currículo

➤ Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Investigadores: Ingresar Currículo”. La única prueba que se puede comprobar es que el usuario ingrese bien la información requerida y que pueda desplazarse con total normalidad por cada una de las pestañas del formulario presentado. El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ Comprobar la Manipulación de Datos

El usuario debe hacer clic cada una de las pestañas del formulario, dar clic en la opción insertar registro, ingresar la información solicitada y dar clic en la opción guardar. Si existe algún error el sistema envía un mensaje.

➤ Condiciones de Ejecución

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y que tenga los permisos respectivos en los menús.

➤ Entrada

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usuario, desplegamos en el menú las opciones: mantenimiento e investigadores.
- Damos clic en ingresar currículo.
- Llenar los datos solicitados en el formulario.
- Guardar la información;

➤ Resultado Esperado

- Que el usuario pueda ingresar correctamente la información.

- Que el usuario pueda moverse por las pestañas del formulario sin ningún inconveniente.

➤ **Evaluación de la Prueba**

- Prueba superada con éxito.

2.6.3 Especificación de Caso de Prueba: Crear Convocatoria

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Convocatorias: Crear Convocatoria”.

Las pruebas a realizar en este caso son:

- Comprobar que el usuario ingrese bien la información.
- Verificar que las listas de valores se muestren correctamente, en especial que la lista de valores del campo formato se muestre de acuerdo al formato seleccionado.

El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usuario, desplegamos en el menú las opciones: mantenimiento, convocatorias y damos clic en la opción crear convocatoria.
- Llenar los datos relevantes del documento y guardar;

➤ **Resultado Esperado**

- Que el usuario pueda ingresar correctamente la información.

➤ **Evaluación de la Prueba**

- Prueba superada con éxito.

2.6.4 Especificación de Caso de Prueba: Asignar Evaluadores

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Convocatorias: Asignar Evaluadores”.

Las pruebas a realizar en este caso son: verificar que las listas de valores se muestren de acuerdo al tipo de evaluador seleccionado y comprobar que el usuario ingrese bien la información. El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: mantenimiento, convocatorias y damos clic en la opción asignar evaluadores.
- Seleccionar el tipo de evaluador: investigador, docente u otro.
- Colocarse en el campo cédula, levantar la lista de valores dando clic en el botón, escoger a la persona que será evaluador, guardar los datos.

➤ **Resultado Esperado**

- Cuando se seleccione el tipo de evaluador, se muestre correctamente la lista de investigadores, docentes u otro.

➤ **Evaluación de la Prueba**

- Se muestra bien la lista de valores y los datos se almacenan correctamente.

2.6.5 Especificación de Caso de Prueba: Asignar Proyectos

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Convocatorias: Asignar Proyectos”.

Las pruebas a realizar en este caso son: verificar que la lista de valores muestre datos con proyectos que estén en el área de investigación del evaluador y que no se asignen proyectos en áreas de investigación equivocadas. El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usuario, desplegamos en el menú las opciones: mantenimiento, convocatorias y damos clic en la opción asignar proyecto evaluador.
- Seleccionar la convocatoria, el investigador y dar clic en el botón listar
- Seleccionar el proyecto de la lista de valores mostrada.

- Guardar la información.
- **Resultado Esperado**
 - Que no se asignen mal los proyectos al evaluador.
- **Evaluación de la Prueba**
 - Se muestra bien la lista de valores y los datos se almacenan correctamente.

2.6.6 Especificación de Caso de Prueba: Registrar Proyecto

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Presentación de Proyectos: Registrar Proyecto”. Las pruebas a realizar en este caso son: verificar que el usuario ingrese la información solicitada, que se desplace de forma normal por las pestañas del formulario y que los botones anexados a algunos campos del formulario trabajen de forma normal. El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usuario, desplegamos en el menú las opciones: procesos, proyectos, presentación y damos clic en la opción ingresar proyecto.
- Seleccionar la convocatoria e ingresar los datos de la pestaña datos generales.
- Utilizar los botones para llenar campos como: localización, dependencia y director.
- Guardar la información y continuar con el resto de pestañas.

➤ **Resultado Esperado**

- Que la información de registro del proyecto de ingrese correctamente y que no queden campos sin ingresar.

➤ **Evaluación de la Prueba**

- Prueba superada correctamente.

2.6.7 Especificación de Caso de Prueba: Evaluar Proyecto

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Evaluación de Proyectos: Evaluar Proyecto”. Las pruebas consisten en verificar la correcta evaluación del proyecto con todos los parámetros de evaluación y que se registre las observaciones realizadas al proyecto. El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: procesos, proyectos, evaluación y damos clic en la opción evaluar proyecto.
- Seleccionar la convocatoria, el evaluador y el proyecto asignado.
- Seleccionar el parámetro, ingresar la nota y presionar el botón calificar parámetro.
- Registrar las observaciones realizadas en el caso de existan.
- Una vez evaluado el proyecto con todos los parámetros presionamos el botón calificar

proyecto.

➤ **Resultado Esperado**

- Que el proyecto sea evaluado satisfactoriamente.

➤ **Evaluación de la Prueba**

- Prueba superada correctamente.

2.6.8 Especificación de Caso de Prueba: Aprobar Proyecto

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Aprobación de Proyectos: Aprobar Proyecto”. Las pruebas consisten en aprobar el proyecto previamente evaluado, anexando una resolución emitida por el Consejo Directivo del CUICYT y que el puntaje obtenido en su evaluación sea el requerido.

El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: procesos, proyectos, aprobación y damos clic en la opción aprobar proyecto.
- Seleccionar la convocatoria y el proyecto.
- En el campo resolución añadimos la resolución existente, dando clic en el botón listar.

- Guardamos los cambios realizados dando clic en el botón aprobar proyecto

➤ **Resultado Esperado**

- Aprobar correctamente los proyectos.

➤ **Evaluación de la Prueba**

- Se supera la prueba a través de la correcta inserción de la resolución desplegada a través de la lista de valores.

2.6.9 Especificación de Caso de Prueba: Priorizar Proyecto

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Oficialización de Proyectos: Priorizar Proyecto”. Las pruebas consisten en asignar el grado de importancia que tiene la ejecución de los proyectos a través de una escala del uno al diez, esta prueba servirá para futuros procesos como la carga de proyectos al módulo de planeamiento estratégico.

El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: procesos, proyectos, oficialización y damos clic en la opción priorizar proyecto.
- Seleccionar la convocatoria y el proyecto.

- Dar clic en el campo prioridad, seleccionar el valor deseado y guardar el cambio realizado dando clic en el botón priorizar proyecto.

➤ **Resultado Esperado**

- Priorizar correctamente el proyecto de acuerdo al grado de importancia del mismo y que se guarde los cambios.

➤ **Evaluación de la Prueba**

- Se supera la prueba ya que el formulario funciona correctamente.

2.6.10 Especificación de Caso de Prueba: Cargar Proyectos POA

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Oficialización de Proyectos: Cargar Proyectos POA”. En este caso de uso se comprobará:

Cargar Proyectos

- Que se muestre en el árbol de datos la lista con todos los proyectos aprobados.
- Que se seleccione correctamente el o los proyectos a cargar.
- Que se utilicen correctamente los botones de: cargar uno, cargar todos, quitar uno o quitar todos, los mismos que son visibles en la pestaña proyectos y actividades.
- Que cuando se seleccione y se cargue un proyecto sus actividades se vean en la pestaña actividades.
- Que se seleccione la o las actividades del proyecto y que se guarden correctamente.
- Que se guarde correctamente los proyectos y actividades módulo de planeamiento estratégico.

Asignar Proyectos POA

- Que los botones de las listas de valores presentados en cada pestaña del formulario muestren datos correctos.
- Que se utilice correctamente la pestaña proyectos y que se ingrese correctamente los datos del proyecto.
- Que se utilice correctamente la pestaña actividades para el ingreso de actividades.
- Que se utilice correctamente la pestaña ítems para el ingreso de requerimientos.
- Que se guarden los cambios realizados en el formulario.

El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

Cargar Proyectos

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: procesos, proyectos, oficialización y damos clic en la cargar proyectos.
- Seleccionar la pestaña proyectos, luego seleccionamos la convocatoria y posteriormente seleccionamos el o los proyectos a cargar.
- Cargar o descargar el o los proyecto utilizando los botones mencionados anteriormente.
- Seleccionar la pestaña actividades, seleccionar la o las actividades a cargar.
- Cargar o descargar las actividades presionando los botones antes mencionados.

Asignar Proyectos POA

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: procesos, proyectos, oficialización y damos clic en la asignar proyectos POA.
- Seleccionar la pestaña proyectos e ingresar la información requerida y guardamos los cambios pulsando la opción guardar.

- Seleccionar la pestaña actividades, llenar los datos solicitados y guardar los cambios dando clic en la opción guardar.
- Seleccionar la pestaña Items, registrar los datos solicitados y guardar la información ingresada con la opción guardar.

➤ **Resultado Esperado**

- Que la información se guarde correctamente.

➤ **Evaluación de la Prueba**

- Prueba superada con éxito.

2.6.11 Especificación de Caso de Prueba: Crear Tipo Informe

➤ **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Ejecución de proyectos: Crear tipo informe”. La única prueba que se puede comprobar es que el usuario ingrese bien la información.

El entorno para realizar la prueba es el formulario de entrada de la aplicación.

➤ **Comprobar la Manipulación de Datos**

El usuario debe hacer clic en la opción insertar registro, ingresar la información solicitada, dar clic en guardar y si existe algún error el sistema envía un mensaje.

➤ **Condiciones de Ejecución**

Las condiciones de ejecución del caso de prueba son: es que el usuario “mgi_pruebas” este dado de alta en la base de datos y tenga los permisos respectivos en los menús.

➤ **Entrada**

- Ingresar al sistema con el usuario “mgi_pruebas” y su respectiva contraseña;
- En el menú principal seleccionamos: inicio, gestión científico- tecnológico y centro de investigación científica tecnológica.
- Aparece la interfaz propia del usurario, desplegamos en el menú las opciones: procesos, proyectos, ejecución y damos clic en informe trimestral.

- Seleccionamos la convocatoria, el proyecto y presionamos el botón crear informe;
- Aparece la pantalla de informes en la que tenemos que ingresar la información del tipo de informe y guardar los datos.
- Ingresar el detalle del informe navegando por cada una de las pestañas que tiene el formulario y guardar los cambios realizados.

➤ **Resultado esperado**

- Que el usuario pueda ingresar correctamente la información.

➤ **Evaluación de la Prueba**

- Prueba superada con éxito.

2.7 LISTA DE RIESGOS

La lista de riesgos del sistema es un compendio de acciones o razones por las cuales el sistema puede experimentar retrasos para así poder establecer un plan de mitigación de riesgos, podrá ser modificada de acuerdo al avance del proyecto y será revisada periódicamente al menos una vez por iteración. A continuación se enumera y detalla cada uno de los riesgos encontrados y se adjuntan las respectivas recomendaciones:

2.7.1 Descripción de los Riesgos

➤ **Diseño Inadecuado**

El diseño del proyecto no es adecuado a la realidad de los procesos existentes.

Medidas de mitigación:

- realizar un análisis profundo del problema, a través de reuniones constantes con el Responsable Funcional y con los expertos en el tema.

➤ **Falta de Experiencia**

La persona responsable del desarrollo del sistema no tiene la suficiente experiencia en la metodología RUP y en la utilización de la plataforma ORACLE ®. Por lo que el “Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa para el CUICYT-UTN” podría retrasarse hasta el mes de noviembre.

Medidas de mitigación:

- Incrementar esfuerzos para el buen desarrollo del sistema.

➤ **Falta de Configuración Óptima de la Base de Datos Oracle**

La configuración de la Base de Datos no es la óptima, el desempeño de la misma depende de una buena distribución de los espacios de memoria, velocidad de respuesta, etc.

Medidas de mitigación:

- hacer evaluaciones constantes del desempeño de la BDD;
- hacer conocer al administrador de la BDD de la institución, sobre el desempeño de la BDD.

➤ **Motivación Económica**

Retrasos en la asignación de recursos económicos al grupo de desarrollo debido a trámites burocráticos.

Medida de mitigación:

- El jefe de proyecto realizara el seguimiento adecuado de los recursos económicos del proyecto.

➤ **Falta de Políticas y Reglamentos**

Algunas actividades que se realizan en la institución no están normadas adecuadamente.

Medidas de mitigación:

- Realizar un estudio adecuado de los reglamentos y políticas que rigen el normal desenvolvimiento la entidad;
- Realizar un análisis profundo de los procesos en la gestión de proyectos;
- En caso de existir procesos que se realizan y que no estén debidamente normadas, la persona responsable del desarrollo del sistema realizara la propuesta adecuada a las autoridades para que estas irregularidades se solucionen y que el proyecto se enmarque dentro de las normas adecuadas.

➤ **Adquisición de Servidores y Equipos de Comunicaciones Retrasada**

La adquisición de servidores y equipos de comunicación no se concreta.

Medidas de mitigación:

- El jefe de proyecto será el encargado de realizar el seguimiento y gestión para la adquisición del hardware necesario para el desarrollo e implementación del proyecto;
- En caso de faltar el hardware necesario, se improvisara configurando e implementando adecuadamente el hardware existente en el departamento.

➤ **Falta de Planificación del Proyecto**

No se realiza la respectiva planificación del sistema a desarrollarse.

Medidas de mitigación:

- El Jefe de proyecto realizará la planificación adecuada para el desarrollo del sistema;
- Se realizará el seguimiento adecuado a la responsable del desarrollo del sistema cada meta se cumpla en los tiempos previstos.

➤ **Los Interesados Desconocen el Proceso de Ingeniería de Software**

Existe poco o ningún conocimiento por parte de los interesados de la institución sobre los procesos que se realizan en la gestión de proyectos.

Medida de mitigación:

- Programar reuniones con los interesados de la institución, haciéndoles conocer la importancia del proyecto y el esfuerzo requerido para su implementación;
- Presentar simulaciones periódicas, las mismas que les dará una clara idea de la importancia del proyecto.

➤ **Falta de Participación del Usuario y Priorización Inadecuada de Requerimientos**

El usuario final no participa de una forma activa en el desarrollo del proyecto.

Medidas de mitigación:

- Planificar reuniones periódicas con los usuarios finales en momentos que no interrumpen su actividad diaria;
- Pacer saber a los usuarios finales del sistema que cualquier aporte por parte de ellos es importante para el desarrollo del proyecto, con esto logramos la participación activa del usuario.

➤ **Planificación Excesivamente Optimista**

No se delimita adecuadamente el alcance del “Sistema para la Gestión de la Investigación Científica, Tecnológica e Innovativa para el CUICYT-UTN”.

Medida de mitigación:

- Realizar el análisis profundo y adecuado para delimitar de una manera real el alcance del sistema.

➤ **Expectativas Irreales**

Crear de falsas expectativas en las autoridades de la institución y usuarios del sistema.

Medidas de mitigación:

- Delimitar adecuadamente el alcance del proyecto;
- Planificar adecuadamente los tiempos de desarrollo.

➤ **Incompatibilidad del Software Cliente**

Incompatibilidad con la Base de Datos y la configuración de clientes finales.

Medidas de mitigación:

- Instalar, configurar y realizar todas las pruebas necesarias en el software cliente;
- Utilizar una sola plataforma en todas las máquinas clientes de la UTN.

CAPÍTULO III

CONCLUSIONES Y RECOMENDACIONES



SISTEMA DE GESTIÓN DE
LA INVESTIGACIÓN
CIENTÍFICA, TECNOLÓGICA
E INNOVATIVA PARA EL
CUICYT - UTN

2.8 CONCLUSIONES

- La Implementación del sistema informático para la Gestión de la Investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN optimiza los procesos, eliminando tareas innecesarias y generando resultados más rápidos.
- El aplicativo desarrollado aporta significativamente al Centro Universitario de Investigación Ciencia y Tecnología agilizando el levantamiento de información que antes no se encontraba disponible.
- La aplicación ofrece una interface amigable al usuario.
- La base de datos Oracle 10g proporciona solidez, confiabilidad, pertenencia, seguridad, integridad, disponibilidad, relevancia con datos almacenados sobre la información del departamento.
- El modelo de datos diseñado para la gestión investigativa permite generar reportes oportunos y necesarios para el CUICYT.
- Al utilizar Tecnología Java para el desarrollo de este aplicativo de software, concluyo que es una herramienta excelente y de gran funcionalidad, además de disminuir costos de licenciamiento, posee características que la hacen solida al manejo de un alto volumen de datos, así también otras características que hacen que estas herramientas estén entre unas de las más recomendada para aplicaciones empresariales.

2.9 RECOMENDACIONES

- El aplicativo desarrollado, de acuerdo a sus características de desarrollo debe ser utilizado en el departamento con el fin de brindar un servicio eficiente y dar a conocer a la ciudadanía los nuevos servicios que puede tener al alcance de sus manos.
- Desarrollar los reportes que sus respectivos usuarios crean convenientes a fin de ofrecer información veraz y confiable.
- Que las autoridades de la Institución se concienticen sobre la importancia de mantener la información actualizada de los procesos académicos investigativos.
- Que la universidad a través del CUICYT genere políticas que garanticen la permanente utilización del sistema implementado para la gestión investigativa.

GLOSARIO DE TÉRMINOS

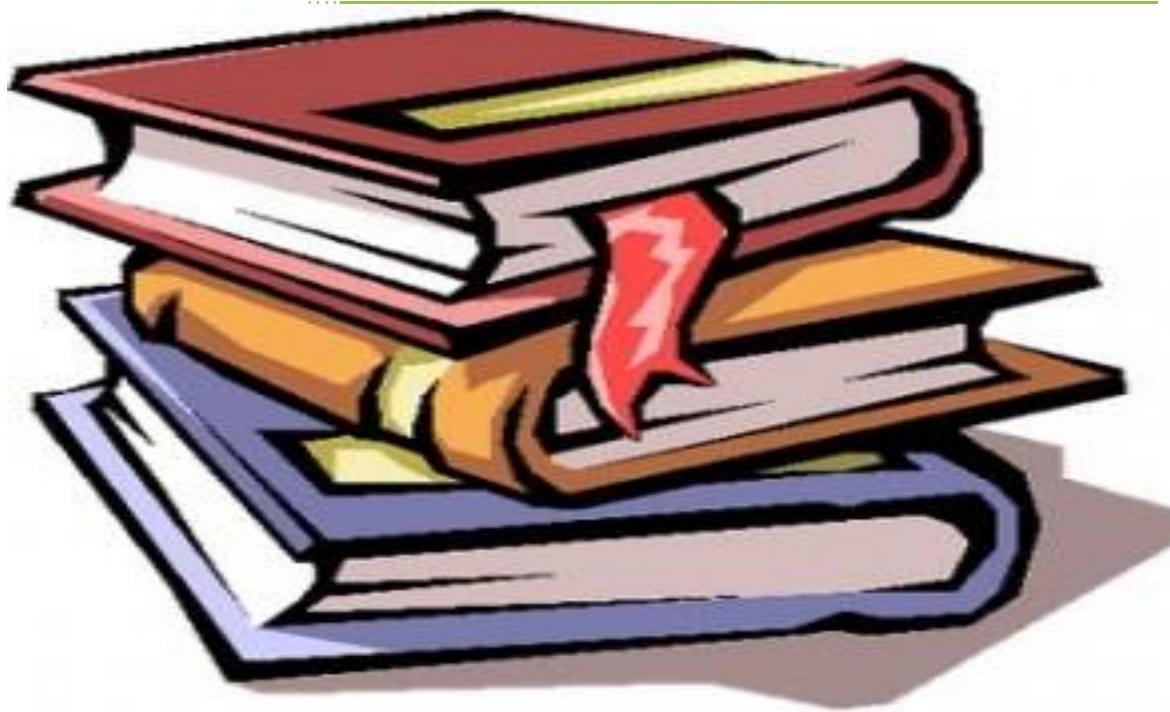


SISTEMA DE GESTIÓN DE
LA INVESTIGACIÓN
CIENTÍFICA, TECNOLÓGICA
E INNOVATIVA PARA EL
CUICYT - UTN

- **UTN:** Universidad Técnica del Norte.
- **CUICYT:** Centro Universitario de Investigación Científica y Tecnológica.
- **Triunvirato:** Gobierno ejercido por tres personas, que normalmente están aliadas.
- **HCR:** Honorable Cámara de Representantes.
- **HCU:** Honorable Consejo Universitario.
- **SENESCYT:** Secretaría Nacional de Educación Superior, Ciencia y Tecnología.
- **Gestión:** A la gestión se la define como acción o trámite que hay que llevar a cabo para conseguir o resolver una cosa. Conjunto de operaciones que se realizan para dirigir y administrar un negocio o una empresa: una buena gestión hace que las empresas ganen dinero.
- **Proyecto:** Un proyecto es un conjunto de actividades relacionadas para lograr un fin específico, con un comienzo y fin claros, sujeto a tres "restricciones" principales: tiempo, presupuesto y alcance.
- **Gestión de proyectos:** La gestión de proyectos también conocida como gerencia, dirección o administración de proyectos es la disciplina de planear, organizar, asegurar y coordinar recursos y personas para cumplir con los objetivos, entregables y criterios de éxito de los proyectos.
- **Suite:** Conjunto de herramientas de software incluidas en un sólo paquete y que se integran entre sí.
- Intervalo de tiempo en el que un programa de computadora se ejecuta en un sistema operativo.
- **RDBMS:** Sistema de Gestión de Base de Datos Relacional.
- **Metadatos:** Son datos que describen otros datos.
- **Memoria:** La memoria interna del ordenador (normalmente la RAM). Un recurso muy importante de las bases de datos, ahí se almacenan los últimos datos utilizados.
- **Procesos:** Programa que se encuentra en ejecución. Un RDBMS lanza muchos procesos que gestionan todos los subsistemas de control necesarios para el buen funcionamiento de la base de datos.
- **SQL:** El lenguaje de consulta más popular de los sistemas de base de datos relacionales.
- **ActiveX:** Es un entorno para definir componentes de software reusables de forma independiente del lenguaje de programación.

- **JavaBeans:** Son un modelo de componentes creado por Sun Microsystems para la construcción de aplicaciones en Java.
- **PLSQL (Procedural Language/Structured Query Language):** Es un lenguaje de programación incrustado en Oracle, soportara todas las consultas, ya que la manipulación de datos que se usa es la misma que en SQL.
- **Sistema:** Conjunto de componentes que se relacionan con al menos algún otro componente para generar uno o más procesos.
- **Backup:** Corresponde a una copia de seguridad de la información original, con el fin de disponer de un medio de recuperarlos en caso de su pérdida.
- **RUP:** Metodología de desarrollo empleada en la ingeniería de software.
- **SGA:** Es la abreviatura de System Global Área, Área Global de Sistema. Está situada al inicio de los datos de la instancia y contiene los datos e información de control de la instancia.
- **PGA:** Zona global de los programas (Program Global Area). En ella se almacenan los datos correspondientes a un proceso (sólo un proceso puede utilizar esta área).
- **Tablespaces:** Pertenecen sólo a una base de datos y sirven para agrupar los datos de la base de datos

BIBLIOGRAFÍA



SISTEMA DE GESTIÓN DE LA
INVESTIGACIÓN
CIENTÍFICA, TECNOLÓGICA
E INNOVATIVA PARA EL
CUICYT - UTN

➤ **Normativas**

^[N1]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Normas de Procedimiento de Investigación**” Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N2]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Del Consejo Directivo**”, Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N3]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Del Director**”, Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N4]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Del Coordinador de Proyectos del CUICYT**”, Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N5]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**De los Investigadores**” Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N6]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Del Apoyo Administrativo**” Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N7]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Normas de Procedimiento de Investigación**” Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N8]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**Proyectos de Investigación**” Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N9]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT).”**De las sanciones**” Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

➤ Páginas Web

[W1] WIKIPEDIA.(s.f.).Oracle Developer Suite. Recuperado el 18 de Diciembre del 2012, de http://en.wikipedia.org/wiki/Oracle_Developer_Suite

[W2] WIKIPEDIA. (s.f.). Oracle Designer. Recuperado el 25 de 11 del 2011, de http://en.wikipedia.org/wiki/Oracle_Designer

[W3] ECURED. (2002). Oracle Designer. Recuperado el 14 de Diciembre del 2012, de http://www.ecured.cu/index.php/Oracle_Designer

[W4] BLOGSPOT. (s.f.). Introducción a Oracle Arquitectura Cliente/Servidor. Recuperado el 28 de Septiembre del 2012, de <http://introduccionaoracle.blogspot.com/p/arquitectura-cliente-servidor.html>

[W5] WIKIPEDIA.(s.f.). Oracle Corporation. Recuperado el 24 de 12 del 2012, de http://es.wikipedia.org/wiki/Oracle_Corporation

[W6] ANÓNIMO. (s.f.). Oracle Components. Recuperado el 14 de Diciembre del 2012, de <http://flylib.com/books/en/3.396.1.23/1/>

[W7] Jesús, V. (Mayo de 1998). Oracle Backup y Recuperación. Recuperado el 15 Septiembre del 2012, de <http://www.infor.uva.es/~jvegas/cursos/bd/oraback/oraback.html>

➤ Documentos Web

[D1] Genero, M. (s.f.). Introducción a Oracle Developer, Recuperado el 28 Octubre del 2012, de <http://es.scribd.com/doc/101126993/Manual-Developer>

[D2] Otiniano, R. (s.f.). Tutorial de Oracle Forms Developer 10g. Recuperado el 3 Junio del 2011, de <http://es.scribd.com/doc/56988520/Oracle-Forms-Developer-10g>

[D3] DASGBD. (2007). Introducción a Designer. Recuperado el 28 de Octubre del 2012, de <http://alarcos.inf-cr.uclm.es/doc/aplicabdd/Documentos/teoria/INTRODUCCIÓN A DESIGNER.pdf>

[D4] DASGBD. (2007). Componentes de Designer, Recuperado el 14 de Diciembre del 2012, de <http://alarcos.inf-cr.uclm.es/doc/aplicabdd/Documentos/teoria/Tema%20-Componetntes%20de%20Designer.pdf>

[D5] Sánchez, J. (2004). Arquitectura del DBMS de Oracle. Recuperado el 18 de Septiembre del 2012, de <http://www.jorgesanchez.net/bd/arquOracle.pdf>

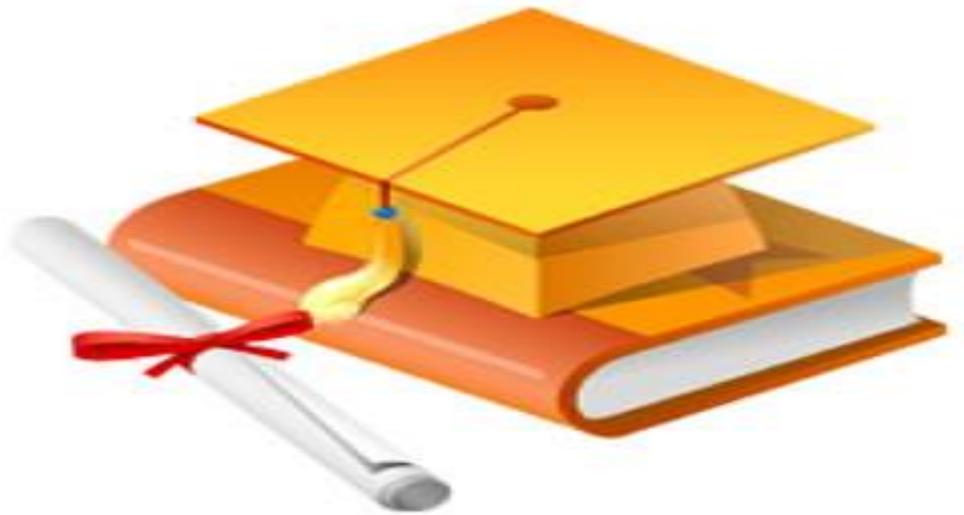
[D6] Riccio, F. (s.f.). Administración de Oracle 10g. Recuperado el 10 de Septiembre del 2012, de <http://www.tinformacion.com/manuales/pdf/administracion-oracle-10g.pdf>

[D7] Paredes, J., Suarez, Y., Aristiy, F. & Sánchez, J. (Julio 2008). Manual de Administración de Base de Datos 1. Recuperado el 06 de Abril del 2012, de http://campus.dokeos.com/courses/BD2USP/document/Manual_de_Administracion_de_Base_de_Datos_Oracle.pdf?cidReq=BD2USP

[D8] Universidad de Jaen. (s.f.). Programación de Base de Datos Oracle. Recuperado el 17 de Agosto del 2012, de http://www.di.ujen.es/~molina/bd1/apuntes_p

[D9] Oracle. (2008). Curso de Administración de Oracle 10g (10.2) Manual del Alumno. Recuperado el 21 de Julio del 2012, de <http://es.scribd.com/doc/19482492/Curso-de-Oracle-10g-Administracion-nivel-basico>

ANEXOS



SISTEMA DE GESTIÓN DE
LA INVESTIGACIÓN
CIENTÍFICA, TECNOLÓGICA
E INNOVATIVA PARA EL
CUICYT - UTN

A. ANTEPROYECTO DE TESIS

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
ESCUELA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

ANTEPROYECTO DE TESIS

DATOS GENERALES

1. TEMA: SISTEMA DE GESTIÓN DE LA INVESTIGACIÓN CIENTÍFICA, TECNOLÓGICA E INNOVATIVA PARA EL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA DE LA UTN.	
2. APLICATIVO: Desarrollo e implementación del sistema informático para la gestión de la investigación científica, tecnológica e innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN.	
3. DISCIPLINA / AREA: Investigación y desarrollo de software.	
4. ENTIDAD QUE AUSPICIA: Centro Universitario de Investigación Científica y Tecnológica de la UTN.	
5. DIRECTOR DEL PROYECTO: Ing. Fernando Garrido Sánchez Msc.	
6. AUTOR: Cristian Patricio Herrera Morales	
7. TELÉFONO: 062 - 918 – 724	
8. CORREO ELECTRÓNICO: cristianpatricio_hm@hotmail.com	
9. DURACIÓN DEL PROYECTO: 12 meses	
10. ESTADO DEL PROYECTO: Nuevo <input checked="" type="checkbox"/> De continuación <input type="checkbox"/>	
11. PRESUPUESTO: ✓ Tesista: 395 USD ✓ Auspiciante: 43.740	
PARA USO DEL CONSEJO ACADÉMICO	
FECHA DE ENTREGA:	FECHA DE REVISIÓN:
APROBADO: SÍ <input type="checkbox"/> NO <input type="checkbox"/>	FECHA DE APROBACIÓN:
OBSERVACIONES:	

PLAN DEL PROYECTO DE TITULACIÓN

Propuesto por: Cristian Herrera	Áreas Técnicas del Tema: ✓ Desarrollo. ✓ Investigación.
Director del Proyecto: Ing. Fernando Garrido Msc.	Fecha: 11 de Abril del 2011.

1. TÍTULO DEL PROYECTO

Sistema de gestión de la investigación científica, tecnológica e innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN.

2. PROBLEMA

2.1 Antecedentes:

La Universidad Técnica del Norte, es una academia de carácter público, tiene como misión esencial contribuir al desarrollo educativo, científico, tecnológico, socioeconómico y cultural de la región norte del país a través de la generación y difusión del conocimiento. Forma profesionales críticos, creativos, capacitados, humanistas y éticos comprometidos con el cambio social y con la preservación del medio ambiente.

La Universidad Técnica del Norte está ubicada en la provincia de Imbabura, cantón Ibarra, barrio El Olivo. La Universidad cuenta con el Centro Universitario de Investigación Científica y Tecnológica "CUICYT" como centro encargado de coordinar la gestión de la investigación que se realiza dentro y fuera de la institución.

La universidad a través del CUICYT desea institucionalizar y sistematizar todo el proceso de investigación en ciencia, tecnología e innovación de manera interactiva y multidisciplinaria para propiciar la creación, adaptación, generación y transferencia de tecnología, en la búsqueda de alternativas viables e innovadoras para la solución de problemas prioritarios de los sectores sociales y productivos de la región norte del país.

El CUICYT a más de impulsar proyectos de investigación es un centro generador de investigadores a través de la investigación en el aula, formando grupos permanentes de estudiantes que se organizan en el aula alrededor de proyectos de investigación, con miras a proponer alternativas de solución a los problemas del entorno; realiza eventos tales como expo-ferias de investigación, talleres, seminarios publicación y difusión de los resultados de las investigaciones.

2.2 Situación Actual:

En la actualidad el CUICYT está realizando el registro de la información (proyectos de investigación, seminarios, talleres, capacitaciones, publicaciones, expo-investigación, etc.) generada en la gestión de la investigación de forma manual; es decir que la documentación de la gestión investigativa que realizan queda archivada en carpetas, lo cual genera muchas limitaciones porque no se tiene un correcto control de esta información y al momento de hacer un levantamiento de la información se emplea mucho tiempo para procesarla ya que la información no está organizada correctamente.

Al no contar con un registro adecuado de la información el centro está limitado a no contar con una fuente de información confiable y oportuna que les permita llevar un control permanente de la gestión investigativa. Todo esto ocasiona que no se pueda hacer un reporte oportuno de cada una de las actividades que realiza el CUICYT.

Esta situación ha limitado que la mayoría de las actividades investigativas que se desarrollan dentro de la universidad y que son gestionadas por el CUICYT no sean conocidas por la comunidad universitaria ni mucho menos por la sociedad.

2.3 Prospectiva:

De continuar con el registro manual de la información que es generada en la gestión de la investigación; para los próximos años el futuro será incierto, ya que han existido reformas aplicadas por el Gobierno Nacional para el tratamiento de la información con lo referente a la investigación que se realiza dentro de las universidades, es por ello que se hace necesario la implementación del sistema para que por medio del Centro Universitario de Investigación Científica y Tecnológica de la Universidad Técnica del Norte, se pueda tener un control oportuno y así evitar la inconsistencia de información, lo cual podría acarrear pérdidas económicas, pérdida de credibilidad y confianza por parte de la sociedad.

2.4 Planteamiento del Problema:

El Centro Universitario de Investigación Científica y Tecnológica registra la información generada en los procesos de gestión de la investigación de forma manual, lo cual provoca problemas tales como pérdida e inseguridad de la información, demora en la elaboración de informes y no se cuenta con información consistente que ayude para la toma de decisiones.

3. OBJETIVOS

3.1 Objetivo General:

- ✓ Diseñar e implementar un sistema de gestión de la investigación científica, tecnológica e innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN que ayude a mantener un control oportuno y permanente de la información.

3.2 Objetivos Específicos:

- ✓ Automatizar los procesos de gestión de la investigación que realiza el CUICYT, mediante un análisis profundo de requerimientos.
- ✓ Elaborar un sistema informático con las debidas seguridades garantizando la disponibilidad e integridad de la información.

- ✓ Desarrollar un sistema informático que sea ágil, amigable y eficiente para mejorar las actividades del quehacer institucional y en especial para el departamento del CUICYT.
- ✓ Garantizar la seguridad de la información generada en el “Sistema de gestión de la investigación científica, tecnológica e innovativa”, a través de los módulos de auditoría y control de usuarios con los que cuenta el sistema integrado de la UTN.
- ✓ Investigar y estudiar el uso de herramientas informáticas y normas de desarrollo de software que se utilizan en la Universidad para garantizar la calidad en el desarrollo informático.
- ✓ Planificar evaluaciones constantes con los usuarios sobre el funcionamiento del sistema en desarrollo, para verificar el cumplimiento de los requerimientos planteados por el CUICYT.
- ✓ Generar información integra, confiable y disponible en cualquier momento.
- ✓ Analizar la integración del “Sistema de gestión de la investigación científica, tecnológica e innovativa” con los sistemas y módulos ya desarrollados en la Universidad para evitar la duplicación de la información.

4. ALCANCE

El sistema informático beneficiará directamente al Centro Universitario de Investigación Científica y Tecnológica, agilizando el trabajo y ofreciendo mejor servicio a los usuarios.

El sistema está diseñado para tener un funcionamiento en modo WEB a través del sistema integrado de la universidad evitando así el aislamiento total de la comunidad universitaria, donde el aplicativo y la base de datos estará instalado en el servidor local y el acceso de los usuarios finales será por la red interna de la universidad y a través de Internet. Cabe destacar que este sistema interactuará con los siguientes sistemas.

FUNCIONAMIENTO INTEGRADO DEL SISTEMA

El sistema de gestión de la investigación científica, tecnológica e innovativa va a interactuar con el sistema académico ya que este maneja información necesaria de las tesis de grado y la gestión académica, el sistema financiero para la asignación de presupuesto para la ejecución de las actividades de investigación, el sistema administrativo a través del módulo de órganos colegiados para conocer las resoluciones emitidas por parte del HCD CUICYT con respecto a la gestión de la investigación, el módulo de convenios para vincular una actividad investigativa con una determinada institución, el módulo de planificación para orientar las actividades de investigación a los objetivos que persigue la UTN y el CUICYT y con el sistema de auditoría y seguridades para mantener un control de la información y la asignación de usuarios



Figura A.3.1. Alcance del Proyecto.
Fuente: [Propia].

SISTEMA DE GESTIÓN, CONTROL Y MANEJO DE PROYECTOS DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

Estudiando las necesidades y requerimientos del CUICYT se ha visualizado de forma clara el correcto y normal funcionamiento del sistema a través de la siguiente definición:

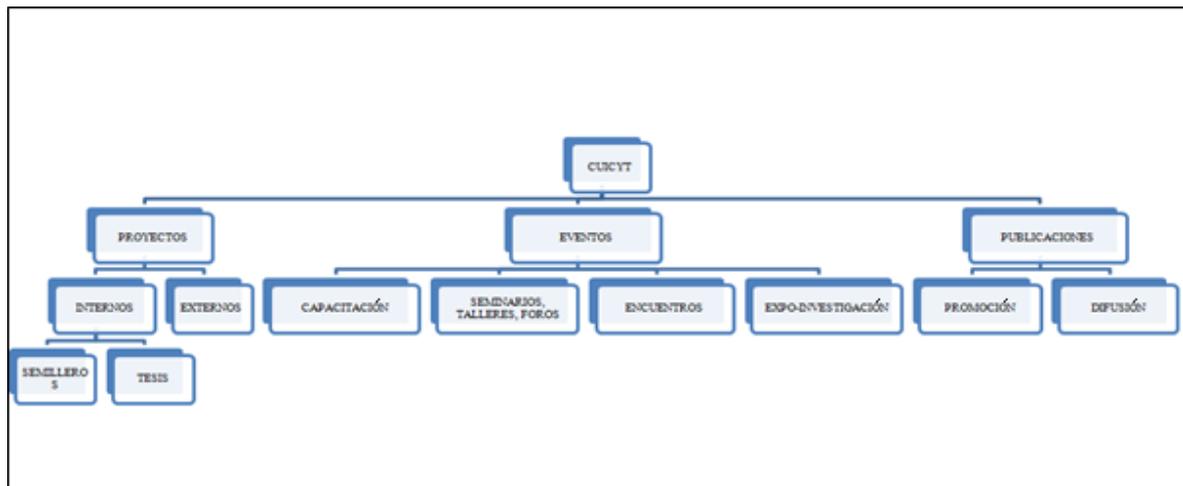


Figura A.3.2. Visión General del Proyecto.
Fuente: [Propia].

Sistema de Gestión, Control y Manejo de Proyectos de Investigación

El sistema tendrá un correcto manejo de los procesos que se manejan para la gestión de proyectos de investigación, eventos y capacitaciones.

El sistema seguirá la secuencia de los siguientes procesos:

Presentación: El docente investigador registra el proyecto planteado en el sistema de acuerdo a las condiciones planteadas por el CUICYT.

Aprobación: El consejo directivo del CUICYT revisa el proyecto y emite una resolución de aprobación del proyecto. Esta resolución será ingresada al sistema de órganos colegiados con el que cuenta la universidad.

Financiamiento: Una vez que se haya aprobado el proyecto se emite una solicitud de asignación de recursos para ejecutar el proyecto, para los cual el sistema trabajara de forma simultánea con el sistema de financiamiento del cual también tiene a su disposición la universidad.

Monitoreo: En el sistema se podrá ingresar los avances del proyectos de investigación tomando en cuenta todos los parámetros de monitoreo que realiza el CUICYT.

Evaluación: El sistema podrá también registrar las evaluaciones pertinentes que se le den a investigación realizada.

Funcionamiento del Módulo de Proyectos de Investigación

El sistema podrá llevar un control permanente de los proyectos de investigación internos, tales como: las tesis de pregrado y postgrado; proyectos de investigación externos, tales como: SENACYT, SEMPLADES, Ministerio del ambiente, municipios, comunidad europea, etc.

Se podrá realizar una correcta administración: ingresar, borrar, actualizar y consultar.

- ✓ Contenido de los proyectos de investigación.
- ✓ Investigadores.
- ✓ Áreas de investigación.
- ✓ Parámetros de evaluación.
- ✓ Financiamiento.

Funcionamiento del Módulo de Eventos

El sistema podrá llevar un control permanente de los eventos que realiza el CUICYT como son las capacitaciones, seminarios, talleres, foros, encuentros, expo-investigación, etc. Mantendrá un registro de las actividades que demanden estos eventos tales como: presupuesto, participantes, ubicación, certificados, etc.

Se podrá realizar una correcta administración: ingresar, borrar, actualizar y consultar.

- ✓ Ingreso de los documentos publicados.

Funcionamiento del Submódulo de Reportes

Cada uno de los módulos explicados contara con un submódulo de reportes, los mismos que estarán ajustados a las necesidades de los usuarios.

- ✓ Reporte de los proyectos de investigación terminados.
- ✓ Reporte de los proyectos de investigación suspendidos.
- ✓ Reporte de eventos realizados.
- ✓ Reporte de publicaciones realizadas.
- ✓ Reporte de los investigadores de los proyectos.
- ✓ Reportes de los participantes de los eventos.

5. JUSTIFICACIÓN

El presente estudio de factibilidad se realiza para determinar la viabilidad del sistema informático, el cual tiene como finalidad automatizar los procesos de gestión de la investigación generada en la universidad e impulsada a través del CUICYT, esto con el objetivo de agilizarlos y así brindar facilidad a los usuarios.

Mediante visitas realizadas al CUICYT se observó la falta de un sistema que les permita a ellos como departamento coordinar, organizar y agilizar los procesos que hasta ahora se los ha realizado manualmente: puesto que existe separación de funciones la información es vulnerable a manipulaciones por parte de cualquier persona y con ello propensa a fraudes de todo tipo, al implementar este sistema informático se pretende agilizar el tiempo de respuesta y garantizar confiabilidad y seguridad en la información.

Para el desarrollo e implementación del sistema se utilizará herramientas financiadas por la universidad las mismas que son requeridas para este tipo de desarrollo y las herramientas a utilizar son las siguientes:

Herramientas de Desarrollo

Hardware

- ✓ Equipos de Computación (para desarrollo)
- ✓ Servidor de Aplicación Web
- ✓ Servidor de Base de Datos
- ✓ Equipo con Web Browser

Software

- ✓ Oracle® Standard One 10g (1 licencia por procesador)
- ✓ Oracle® Developer Suite Release 10g
- ✓ Oracle® Application Server 10g,
- ✓ Oracle® Forms. Repon Server (1 licencia por procesador)

Arquitectura

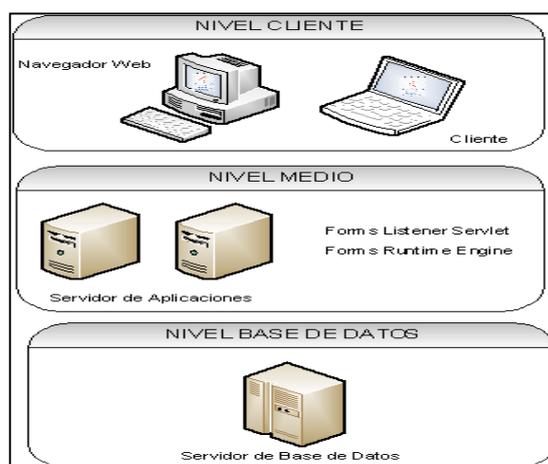


Figura A.3.3. Arquitectura.
Fuente: [Propia].

Metodología de Desarrollo

Racional Unificad (RUP)

Es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Se usará la metodología de desarrollo Proceso Racional Unificad (RUP) en sus cuatro etapas:



Figura A.3.4. Fases Metodología de Desarrollo.
Fuente: [Propia].

- ✓ **Fase Inicial:** comprensión del problema y la tecnología. Delimitación del ámbito o alcance del sistema. Establecimiento de una línea base de la arquitectura.
- ✓ **Elaboración:** Construir una versión ejecutable de la arquitectura y la aplicación. Trabajo en requerimientos, modelo de negocios, análisis y diseño.
- ✓ **Construcción:** Completar el esqueleto de la aplicación con la funcionalidad. Construir una versión beta.
- ✓ **Transición:** Hacer disponible la aplicación para los usuarios finales. Construir la versión Final.

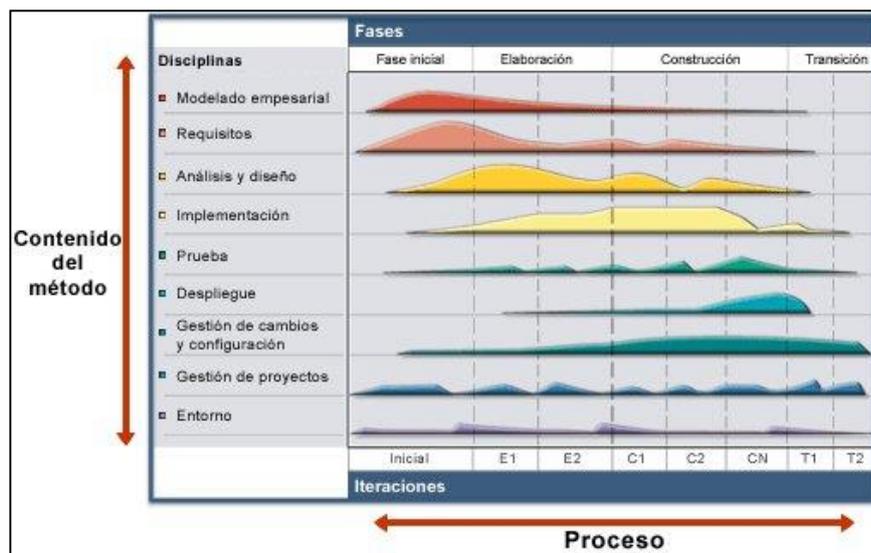


Figura A.3.5. Metodología RUP.
Fuente: [Propia].

6. PRESUPUESTO

DETALLE		COSTO AUSPICIANTE	COSTO TESISTA	COSTO REAL
Hardware	Equipos de Computación	1'500.00	0.00	0.00
	Servidor de Aplicación Web.	5'000.00	0.00	0.00
	Servidor de Base de Datos.	5'000.00	0.00	0.00
	Equipo con Web Browser.	700.00	0.00	0.00
Software	Oracle® Standard One 10g (1 licencia por procesador).	5'000.00	0.00	0.00
	Oracle® Developer Suite Release 10g.	5'000.00	0.00	0.00
	Oracle® Application Server 10g, Oracle® Forms, Report Server (1 licencia por procesador).	20'000.00	0.00	0.00
Costo de Desarrollo	Pago por desarrollo	1'000.00	0.00	0.00
Capacitación a los desarrollares	Cursos y Libros, Asesoramiento.	800.00	0.00	0.00
Proyecto	Papelería y Suministros de Oficina.	500.00	395.00	395.00
Subtotal	(Parcial).	43'500.00	0.00	0.00
5% Imprevistos		240.00	0.00	0.00
Total		43'740.00	395.00	395.00

Tabla A.1. Presupuesto.

Fuente: [Propia].

7. TEMAS AFINES REALIZADOS

N.º	TÍTULO	AUTOR	AÑO	SEMEJANZA	DIFERENCIA
1	Sistema Gerencial Informático para la Gestión y Desarrollo de la Investigación, para el Centro Universitario de Investigación Científica y Tecnológica	Iván Vinicio Chiles Valencia	2001	En esta tesis se realiza un análisis de los procesos de gestión y desarrollo de los proyectos de investigación.	El Sistema de gestión, manejo y control de proyectos de investigación pretende tener un control permanente de los proyectos que

	(CUICYT).				ejecuta el CUICYT.
2	Análisis y Estudio del sistema de Gestión de la Seguridad de la Información (SGSI) para la Universidad Técnica del Norte.	Burgos Yar Verónica Viviana	2009	Se ayudará de este sistema ya que se utilizará la resolución emitida por el HCU para el CUICYT con respecto a un proyecto de investigación. Además este sistema utiliza la herramienta de desarrollo Oracle forms.	El sistema mantendrá un control de los proyectos de investigación; evitando de esta forma que queden solamente como una resolución, sino más bien que estos sean ejecutados y que la información que se genere se a accesible a los usuarios pertinentes.
3	Sistema de administración y gestión de recursos humanos para la Universidad Técnica del Norte.	Gissela Elizabeth Muñoz Padilla	2009	El uso de Oracle forms como plataforma de desarrollo para la gestión del recurso humano.	E sistema se centrará en un sistema de control de proyectos de investigación y no de recursos humanos.

Tabla A.2. Temas Afines.

Fuente: [Propia].

8. TEMARIO

CAPÍTULO 1:

INTRODUCCIÓN

- ✓ Generalidades de la Empresa.
- ✓ Organización de la Empresa.
- ✓ Situación Actual de la Empresa.
- ✓ Diagnóstico de la Empresa.

CAPÍTULO 2:

MARCO TEÓRICO

- ✓ Sistemas operativos (DISPONIBILIDAD PARA VARIOS SISTEMAS OPERATIVOS).
- ✓ Bases de datos.
- ✓ Factibilidad de plataformas.
- ✓ Leyes y reformas para la educación superior.
- ✓ Servidores de aplicaciones Web.
- ✓ Herramientas de reportes.

CAPÍTULO 3:

ARQUITECTURA DEL SISTEMA

- ✓ Funcionamiento del Sistema.
- ✓ Definición de los módulos.
- ✓ Funcionamiento de los módulos.

CAPÍTULO 4:

DISEÑO Y DESARROLLO DEL APLICATIVO

- ✓ Incepción.
- ✓ Elaboración.
- ✓ Construcción.
- ✓ Transición.

CAPÍTULO 5:

CONCLUSIONES Y RECOMENDACIONES

- ✓ Conclusiones.
- ✓ Recomendaciones.
- ✓ Anexos
- ✓ Glosario.
- ✓ Manual técnico.
- ✓ Manual de Usuario.
- ✓ Bibliografía.

9. CRONOGRAMA DE ACTIVIDADES

N	NOMBRE DE TAREA	DURACIÓN	M 1	M 2	M 3	M 4	M 5	M 6	M 7	M 8	M 9	M 10	M 11	M 12
1	Recopilar Información	60 días												
2	Estudio Oracle® Standard One 10g	20 días												
3	Estudio Oracle® Developer Suite Release 10g	20 días												
4	Estudio Oracle® Application Server 10g	20 días												
5	Estudio Oracle® Forms, Report	20 días												

	Server																		
6	Estudio de herramientas adicionales "Toad"	10 días																	
7	Estudio de las fases RUP	Permanente																	
8	Desarrollo del Aplicativo	150 días																	
9	Pruebas locales	15 días																	
10	Corrección de posibles errores	15 días																	
11	Instalación y configuración de Servidor	5 días																	
12	Capacitación a los usuarios	5 días																	
13	Documentación	Permanente																	
14	Entrega del Proyecto	15 días																	

Tabla A.3. Cronograma de Actividades.

Fuente: [Propia].

10. BIBLIOGRAFÍA

10.1 LIBROS y MANUALES

- ✓ Tesis de Sistema Gerencial Informático para la Gestión y Desarrollo de la Investigación, para el Centro Universitario de Investigación Científica y Tecnológico, 2001.

10.2 DIRECCIONES ELECTRÓNICAS

- ✓ La Informática Aplicada a la Administración ,1999,
<http://www.monografias.com/trabajos16/sistemas-informacion-empresa/sistemas-informacion-empresa.shtml>
- ✓ Oracle Overview of Forms Server, 2010,
http://www.uriit.ru/japan/Our_Resources/Doc_iAS/forms.6i/a83591/chap02
- ✓ RUP Etapa diseño, 2009,
<http://www.scribd.com/doc/395783/RUP-etapa-diseno>
- ✓ Documentación general de Oracle
<http://www.oracle.com/index.html>

B. DICCIONARIO DE DATOS

B.1. NOMBRE DE LAS TABLAS DEL SISTEMA

- MGI_TAB_ACTIVIDAD_EJECUTADA.
- MGI_TAB_AREA_INVE_PROYECTO.
- MGI_TAB_AREA_INVESTIGACION.
- MGI_TAB_AREINV_LININV_EJETRA.
- MGI_TAB_ARTICULOS_CIENTIFICOS.
- MGI_TAB_ASISTENTES.
- MGI_TAB_AVANCE_OBJETIVOS.
- MGI_TAB_BENEFICIARIOS.
- MGI_TAB_CATEGORIA_PROYECTO.
- MGI_TAB_CONCLUSIONES_Y_RECOMEN.
- MGI_TAB_CONTENIDO_EVENTO.
- MGI_TAB_CONV_AREA_INV.
- MGI_TAB_CONV_LIN_INV.
- MGI_TAB_CONV_POL_INV.
- MGI_TAB_CONVOCATORIA.
- MGI_TAB_CURSOS.
- MGI_TAB_DEFINICION_INV.
- MGI_TAB_DOCUMENTOS_RESPALDO.
- MGI_TAB_EJES_TRANSVERSALES.
- MGI_TAB_EQUIPO_TRABAJO.
- MGI_TAB_ESTADO_INFORME.
- MGI_TAB_ESTADO_PROYECTO.
- MGI_TAB_ESTRUCTURA_FORMATO.
- MGI_TAB_EVALUACION_PROYECTO.
- MGI_TAB_EVALUADORES.

- MGI_TAB_EVENTOS.
- MGI_TAB_EXPERIENCIA_INVESTIGAC.
- MGI_TAB_EXPOSICIONES_PRESENTAD.
- MGI_TAB_FACILITADORES.
- MGI_TAB_FORMACION_ACADEMICA.
- MGI_TAB_FUENTE_VERIFICACION.
- MGI_TAB_HORARIO_TRABAJO.
- MGI_TAB_INDICADOR.
- MGI_TAB_INFORME.
- MGI_TAB_INSTITUCION_ASOCIADA.
- MGI_TAB_INVESTIGADORES.
- MGI_TAB_LINEA_BASE.
- MGI_TAB_LINEA_INVE_PROYECTO.
- MGI_TAB_LINEA_INVESTIGACION.
- MGI_TAB_MARCO_LOGICO.
- MGI_TAB_MES_PERIODO.
- MGI_TAB_NIVEL_PARTICIPACION.
- MGI_TAB_OBSERVACIONES.
- MGI_TAB_PARAMETROS_EVALUACION.
- MGI_TAB_PARAMETROS_PROYECTO.
- MGI_TAB_PATENTES.
- MGI_TAB_PERFIL_PROYECTO.
- MGI_TAB_PERIODO.
- MGI_TAB_PERIODO_PROYECTO.
- MGI_TAB_POL_INVESTIGACION.
- MGI_TAB_PORCENTAJE_EVALUACION.
- MGI_TAB_PROBLEMAS.
- MGI_TAB_PROCEDIMIENTO_CONV.

- MGI_TAB_PRODUCION_CIENTIFICA.
- MGI_TAB_PROMEDIO_PARAMETRO.
- MGI_TAB_PROYECTO_EVALUADOR.
- MGI_TAB_PROYECTOS.
- MGI_TAB_PROYECTOS_RELACIONADOS.
- MGI_TAB_PUBLICACIONES
- MGI_TAB_RESULTADOS.
- MGI_TAB_RUBRO_MES.
- MGI_TAB_RUBRO_PERIODO.
- MGI_TAB_RUBRO_PERIODO_DET.
- MGI_TAB_SECT_IMPA_PROYECTOS.
- MGI_TAB_SECTOR_IMPACTO.
- MGI_TAB_SUPUESTOS.
- MGI_TAB_TIPO_CONVOCATORIA.
- MGI_TAB_TIPO_DOCUMENTO_RESP.
- MGI_TAB_TIPO_EVENTO.
- MGI_TAB_TIPO_FORMATO.
- MGI_TAB_TIPO_INFORME.
- MGI_TAB_TIPO_LOCALIDAD.
- MGI_TAB_TIPO_OBJETIVO.
- MGI_TAB_TIPO_PARTICIPACION.
- MGI_TAB_TIPO_PROYECTO.
- MGI_TAB_TIPO_RUBRO.
- MGI_TAB_TIPOLOGIA.
- MGI_TAB_TRANSFERENCIA.
- MGI_TAB_USUARIOS.
- MGI_TAB_VIGENCIA_FORMATO.

B.2. EXPLICACIÓN DE LOS CAMPOS DE CADA TABLA

➤ Tabla: MGI_TAB_ACTIVIDAD_EJECUTADA

Entidad que almacena las actividades ejecutadas en los proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe
INVESTIGADOR_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
REQUIERE_PROGRAMACION	Y	VARCHAR2 (1 Byte)	Corresponde a si requiere Programación o no

Tabla B.1. Atributos de la tabla MGI_TAB_ACTIVIDAD_EJECUTADA.

Fuente: [Propia].

➤ Tabla: MGI_TAB_AREA_INVE_PROYECTO

Entidad que almacena el o las áreas de investigación en las que se encuentra perfilado el proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
POLITICA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la política de investigación
AREA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a una observación

Tabla B.2. Atributos de la tabla MGI_TAB_AREA_INVE_PROYECTO.

Fuente: [Propia].

➤ Tabla: MGI_TAB_AREA_INVESTIGACION

Entidad que almacena las áreas de investigación de acuerdo a un tipo de formato.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	N	VARCHAR2 (100 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación
TIPO_FORM_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código tipo de formato
FECHA_VIGENCIA	Y	DATE	Corresponde a la fecha de vigencia

Tabla B.3. Atributos de la tabla MGI_TAB_AREA_INVESTIGACION.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_AREINV_LININV_EJETRA**

Entidad que almacena las áreas de investigación de acuerdo a los ejes transversales.

Atributo	¿Nulo?	Tipo Dato	Descripción
EJE_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del eje transversal
AREA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
LINEA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la línea de investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.4. Atributos de la tabla MGI_TAB_AREINVE_LININV_EJETRA.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_ARTICULOS_CIENTIFICOS**

Entidad que almacena información de los artículos científicos generados como producto de una investigación o ejecución de un proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
TITULO	Y	VARCHAR2 (200 Byte)	Corresponde al título
PRESENTADO_PUBLICAR	Y	VARCHAR2 (1 Byte)	Corresponde a si esta presentado para publicar
REGISTRADO_LATINDEX	Y	VARCHAR2 (1 Byte)	Corresponde a si está registrado
REVISTA_EDITORIAL	Y	VARCHAR2 (20 Byte)	Corresponde a la revista
FECHA_PUBLICACION	Y	DATE	Corresponde a la fecha de publicación
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe

Tabla B.5. Atributos de la tabla MGI_TAB_ARTICULOS_CIENTIFICOS.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_ASISTENTES**

Entidad que almacena información de las personas que asisten a un evento que se realiza como producto de una investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
EVENTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del evento
PERSONA_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula de la persona
DOCENTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del docente
ESTUDIANTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del estudiante
INVESTIGADOR_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.6. Atributos de la tabla MGI_TAB_ASISTENTES.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_AVANCE_OBJETIVOS**

Entidad que almacena información del avance de los objetivos del proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
IMPACTO	Y	VARCHAR2 (20 Byte)	Corresponde al impacto
PORCENTAJE_PARCIAL	Y	NUMBER	Corresponde al porcentaje parcial
PORCENTAJE_ACUMULADO	Y	NUMBER	Corresponde al porcentaje acumulado
MONTO_PRESUPUESTADO	Y	NUMBER	Corresponde al monto presupuestado
MONTO_ACUMULADO	Y	NUMBER	Corresponde al monto acumulado
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe
MARCO_LOGICO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del marco lógico
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.7. Atributos de la tabla MGI_TAB_AVANCE_OBJETIVOS.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_BENEFICIARIOS**

Entidad que almacena información de los beneficiarios directos e indirectos del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
NUMERO	Y	NUMBER	Corresponde al número
SEXO	Y	VARCHAR2 (1 Byte)	Corresponde al sexo
URBANO_RURAL	Y	VARCHAR2 (10 Byte)	Corresponde al sector rural o urbano
TIPO_BENEFICIARIO	Y	VARCHAR2 (10 Byte)	Corresponde al tipo de beneficiario
DIRECTO_INDIRECTO	Y	VARCHAR2 (10 Byte)	Corresponde a si es beneficiario directo o indirecto
APORTE_ECONOMICO	Y	NUMBER	Corresponde al aporte económico
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.8. Atributos de la tabla MGI_TAB_BENEFICIARIOS.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CATEGORIA_PROYECTO**

Entidad que almacena información de la categoría del proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.9. Atributos de la tabla MGI_TAB_CATEGORIA_PROYECTO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CONCLUSIONES_Y_RECOMEN**

Entidad que almacena información de las conclusiones y recomendaciones realizadas en la presentación de un informe.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
TIPO	Y	VARCHAR2 (20 Byte)	Corresponde al tipo
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe

Tabla B.10. Atributos de la tabla MGI_TAB_CONCLUSIONES_Y_RECOMEN.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CONTENIDO_EVENTO**

Entidad que almacena información del contenido del evento que se genera como producto de una investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
NUMERO	Y	NUMBER	Corresponde al número
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
DURACION	Y	VARCHAR2 (20 Byte)	Corresponde a la duración
FECHA	Y	DATE	Corresponde a la fecha
EVENTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del evento
FACILITADOR_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del facilitador
RESPONSABLE	Y	VARCHAR2 (100 Byte)	Corresponde al responsable

Tabla B.11. Atributos de la tabla MGI_TAB_CONTENIDO_EVENTO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CONV_AREA_INV**

Entidad que almacena información de las áreas de investigación bajo las que se realiza una convocatoria a presentación de proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
POLITICA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la política de investigación
AREA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.12. Atributos de la tabla MGI_TAB_CONV_AREA_INV.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CONV_LIN_INV**

Entidad que almacena información de las líneas de investigación bajo las que se realiza una convocatoria a presentación de proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Código de la convocatoria
POLITICA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la política de investigación
AREA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
LINEA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la línea de investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.13. Atributos de la tabla MGI_TAB_CONV_LIN_INV.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CONV_POL_INV**

Entidad que almacena información de las políticas de investigación bajo las que se realiza una convocatoria a presentación de proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Código de la convocatoria
POLITICA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Código política de investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.14. Atributos de la tabla MGI_TAB_CONV_POL_INV.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CONVOCATORIA**

Entidad que almacena información de los datos de una convocatoria realizada, ya sea esta para presentación de proyectos o para eventos.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Estado de la convocatoria
CODIGO	N	VARCHAR2 (10 Byte)	Código de la convocatoria
TEMA	Y	VARCHAR2 (100 Byte)	Tema de la convocatoria
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
FECHA_INICIAL	Y	DATE	Fecha de la convocatoria
FECHA_FINAL	Y	DATE	Fecha fin convocatoria
RESPONSABLE	Y	VARCHAR2 (200 Byte)	Responsable de la convocatoria
LUGAR	Y	VARCHAR2 (100 Byte)	Lugar de la convocatoria
VIGENCIA_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de vigencia
FORMATO_CODIGO	Y	VARCHAR2 (10 Byte)	Código formato
TIPO_CONV_CODIGO	Y	VARCHAR2 (10 Byte)	Código del tipo de convocatoria
TIPO_FORM_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al tipo de formato
PER_ACAD_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al período académico

Tabla B.15. Atributos de la tabla MGI_TAB_CONVOCATORIA.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_CURSOS**

Entidad que almacena información de los cursos que ha realizado un investigador.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
INSTITUCION	Y	VARCHAR2 (50 Byte)	Corresponde a la institución
DURACION	Y	VARCHAR2 (20 Byte)	Corresponde a la duración
ANIO	Y	VARCHAR2 (10 Byte)	Corresponde al año
CERTIFICADO_OBTENIDO	Y	VARCHAR2 (50 Byte)	Corresponde al certificado obtenido
PERSONA_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.16. Atributos de la tabla MGI_TAB_CURSOS.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_DEFINICION_INV**

Entidad que almacena información de la definición de la investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.17. Atributos de la tabla MGI_TAB_CURSOS.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_DOCUMENTOS_RESPALDO**

Entidad que almacena información de los documentos de respaldo, los mismos que pueden ser: oficio, formato proyecto, convocatoria, etc.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (20 Byte)	Corresponde al código
CODIGO_DOC	Y	VARCHAR2 (10 Byte)	Corresponde al código del documento de respaldo
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
DOCUMENTO	Y	BLOB	Corresponde al documento
TIPO	Y	VARCHAR2 (10 Byte)	Corresponde al tipo
FECHA	Y	DATE	Corresponde a la fecha
TIPO_DOCU_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del tipo de documento

Tabla B.18. Atributos de la tabla MGI_TAB_DOCUMENTOS_RESPALDO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EJES_TRANSVERSALES**

Entidad que almacena los ejes transversales de la investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.19. Atributos de la tabla MGI_TAB_EJES_TRANSVERSALES.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EQUIPO_TRABAJO**

Entidad que almacena información del equipo de trabajo que tendrá el proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
DOCENTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del docente
EMPLEADO_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del empleado
ESTUDIANTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del estudiante
INVESTIGADOR_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
TIPO_PART_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al tipo de participación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación
PERSONA_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula de la persona

Tabla B.20. Atributos de la tabla MGI_TAB_EQUIPO_TRABAJO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_ESTADO_INFORME**

Entidad que almacena información de los diferentes tipos de estado en los que un informe de avance del proyecto puede encontrarse.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.21. Atributos de la tabla MGI_TAB_ESTADO_INFORME.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_ESTADO_PROYECTO**

Entidad que almacena información de los diferentes tipos de estado en los que un proyecto puede encontrarse.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.22. Atributos de la tabla MGI_TAB_ESTADO_PROYECTO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_ESTRUCTURA_FORMATO**

Entidad que almacena información de la estructura de un formato de proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde a la descripción
CODIGO_HIJO	Y	VARCHAR2 (10 Byte)	Corresponde al código del hijo
TIPO_FORM_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del tipo de formato
TIPO_FORM_HIJO	Y	VARCHAR2 (10 Byte)	Corresponde al código del tipo de formato hijo
VIGENCIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la vigencia del formato
VIGENCIA_HIJO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la vigencia hijo
OBSERVACION	Y	VARCHAR2 (2000 Byte)	Corresponde a la observación

Tabla B.23. Atributos de la tabla MGI_TAB_ESTRUCTURA_FORMATO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EVALUACION_PROYECTO**

Entidad que almacena información de la evaluación realizada a un proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
NOTA	Y	NUMBER	Corresponde a la nota obtenida
NOTA_FINAL	Y	NUMBER	Corresponde a la nota final
FECHA_EVALUACION	Y	DATE	Corresponde a la fecha de evaluación
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Código de la convocatoria
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
TIPPRO_CODIGO	N	VARCHAR2 (10 Byte)	Código del tipo de proyecto
EVALUADOR_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del evaluador
PARAMETRO_PROY_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del parámetro del proyecto
PARAMETRO_EVAL_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del parámetro de evaluación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.24. Atributos de la tabla MGI_TAB_EVALUACION_PROYECTO.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EVALUADORES**

Entidad que almacena información de las personas encargadas de evaluar los proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
CONVOCATORIA_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
INVESTIGADOR_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
DOCENTE_CEDULA	Y	VARCHAR2 (10 Byte)	Corresponde a la cédula del docente
PERSONA_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula de la persona
AREA_INV_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
OBSERVACION	Y	VARCHAR2 (300 Byte)	Corresponde al código de la observación

Tabla B.25. Atributos de la tabla MGI_TAB_EVALUADORES.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EVENTOS**

Entidad que almacena información de los eventos originados como resultado de una investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde a la descripción
DIRIGIDO_A	Y	VARCHAR2 (100 Byte)	Corresponde a quien está dirigido
FECHA	Y	VARCHAR2 (100 Byte)	Corresponde a la fecha
DURACION	Y	VARCHAR2 (50 Byte)	Corresponde a la duración
COSTO	Y	NUMBER	Corresponde al costo
LUGAR	Y	VARCHAR2 (100 Byte)	Corresponde al lugar
HORARIO	Y	VARCHAR2 (100 Byte)	Corresponde al horario
INSCRIPCIONES	Y	VARCHAR2 (100 Byte)	Corresponde al lugar de las inscripciones
PROMOCION	Y	VARCHAR2 (4000 Byte)	Corresponde al medio de publicación del evento
CUPO_LIMITADO	Y	NUMBER	Corresponde al número de cupos
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
TIPO_EVEN_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de tipo de evento
RESPONSABLE	Y	VARCHAR2 (150 Byte)	Corresponde al nombre del responsable del evento
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
FACILITADOR	Y	VARCHAR2 (4000 Byte)	Corresponde al nombre del facilitador

PARTICIPANTES	Y	VARCHAR2 (4000 Byte)	Corresponde a los participantes
LOGISTICA	Y	VARCHAR2 (4000 Byte)	Corresponde a la logística
ECONOMICA	Y	VARCHAR2 (4000 Byte)	Corresponde al presupuesto
EVALUACION	Y	VARCHAR2 (4000 Byte)	Corresponde a la evaluación

Tabla B.26. Atributos de la tabla MGI_TAB_EVENTOS.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EXPERIENCIA_INVESTIGAC**

Entidad que guarda información de la experiencia en investigación adquirida por los investigador durante su vida profesional, estos datos están relacionado con el currículum del investigador.

Atributo	¿Nulo?	Tipo Dato	Descripción
INSTITUCION	Y	VARCHAR2 (50 Byte)	Corresponde a la institución en la que trabajó
TEMATICA	Y	VARCHAR2 (200 Byte)	Corresponde a la temática
DESDE	Y	DATE	Corresponde a la fecha en que inicia
HASTA	Y	DATE	Corresponde a la fecha en que termina
PERSONA_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde al cédula del investigador
AREA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
CARGO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del cargo
LOCALIDAD_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la localidad
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.27. Atributos de la tabla MGI_TAB_EXPERIENCIA_INVESTIGAC.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_EXPOSICIONES_PRESENTAD**

Entidad que almacena información de las exposiciones presentadas por los investigadores como resultado de una investigación realizada.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
TITULO	Y	VARCHAR2 (100 Byte)	Corresponde al título
LUGAR	Y	VARCHAR2 (50 Byte)	Corresponde al lugar
FECHA	Y	DATE	Corresponde a la fecha de la exposición
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe

Tabla B.28. Atributos de la tabla MGI_TAB_EXPOSICIONES_PRESENTAD.

Fuente: [Propia].

➤ **Tabla: MGI_TAB_FACILITADORES**

Entidad que almacena información de las personas que actuarán como facilitadores de algún evento realizado como resultado de una investigación realizada.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
EVENTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del evento
PERSONA_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula de la persona que actuará como facilitador
DOCENTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del docente que actuará como facilitador
INVESTIGADOR_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.29. Atributos de la tabla MGI_TAB_FACILITADORES.

Fuente: [Propia].

➤ MGI_TAB_FORMACION_ACADEMICA

Entidad que almacena información de la formación académica de los investigadores, esta información está relacionada con el currículum del investigador.

Atributo	¿Nulo?	Tipo Dato	Descripción
DURACION	Y	VARCHAR2 (20 Byte)	Corresponde a la duración
ANIO	Y	VARCHAR2 (20 Byte)	Corresponde al año de graduación
TITULO	N	VARCHAR2 (100 Byte)	Corresponde al título obtenido
PERSONA_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
NIV_EDU_CODIGO	N	NUMBER	Corresponde al nivel de educación
UNIVERSIDAD_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la universidad en que estudió
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a una observación

Tabla B.30. Atributos de la tabla MGI_TAB_FORMACION_ACADEMICA.

Fuente: [Propia].

➤ MGI_TAB_FUENTE_VERIFICACION

Entidad que guarda información de la fuente de verificación que forma parte del marco lógico del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
MARCO_LOG_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del marco lógico
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.31. Atributos de la tabla MGI_TAB_FUENTE_VERIFICACION.

Fuente: [Propia].

➤ MGI_TAB_HORARIO_TRABAJO

Entidad que almacena información del horario de trabajo que tienen los investigadores.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
HORA_INICIO	N	VARCHAR2 (10 Byte)	Corresponde a la hora de inicio
HORA_FIN	N	VARCHAR2 (10 Byte)	Corresponde a la hora en que termina
LUNES	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día lunes
MARTES	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día martes
MIERCOLES	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día miércoles
JUEVES	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día jueves
VIERNES	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día viernes
SABADO	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día sábado
DOMINGO	Y	VARCHAR2 (1 Byte)	Corresponde a la jornada del día domingo
INVESTIGADOR_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.32. Atributos de la tabla MGI_TAB_HORARIO_TRABAJO.

Fuente: [Propia].

➤ MGI_TAB_INDICADOR

Entidad que almacena información de los indicadores que forman parte del marco lógico del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
MARCO_LOG_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del marco lógico
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.33. Atributos de la tabla MGI_TAB_IDICADOR.

Fuente: [Propia].

➤ MGI_TAB_INFORME

Entidad que almacena información de los informes de avance de los proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
EMPLEADO_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del empleado
ESTUDIANTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del estudiante
DOCENTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del docente
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación
ESTADO_INF_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al estado del informe
TIPO_INF_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al tipo de informe
INVESTIGADOR_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador

PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
FECHA_PRESENTACION	Y	DATE	Corresponde a la fecha de presentación
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde a la descripción
NUMERO	Y	NUMBER	Corresponde al número de informe
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código

Tabla B.34. Atributos de la tabla MGI_TAB_IDICADOR.

Fuente: [Propia].

➤ MGI_TAB_INSTITUCION_ASOCIADA

Entidad que almacena información de las instituciones que están asociadas al proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
SIGLAS	Y	VARCHAR2 (50 Byte)	Corresponde a las siglas
NOMBRE	Y	VARCHAR2 (100 Byte)	Corresponde al nombre
DIRECCION	Y	VARCHAR2 (50 Byte)	Corresponde a la dirección
TELEFONO	Y	VARCHAR2 (10 Byte)	Corresponde al número de teléfono
FAX	Y	VARCHAR2 (50 Byte)	Corresponde al número de fax
CORREO_ELECTRONICO	Y	VARCHAR2 (50 Byte)	Corresponde a la dirección de correo
PAGINA_WEB	Y	VARCHAR2 (50 Byte)	Corresponde a la dirección web
MONTO_ASIGNADO	Y	NUMBER	Corresponde al monto asignado
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
LOCALIDAD_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la localidad
NIVEL_PART_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al nivel de participación
CONVENIO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del convenio
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.35. Atributos de la tabla MGI_TAB_INSTITUCION_ASOCIADA.

Fuente: [Propia].

➤ MGI_TAB_INVESTIGADORES

Entidad que almacena información de los investigadores.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
FECHA_INICIO	Y	DATE	Corresponde a la fecha de inicio
FECHA_FIN	Y	DATE	Corresponde a la fecha de finalización
HORAS_DEDICACION	Y	NUMBER	Corresponde al número de horas
DOCENTE_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
TIPO_PART_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al tipo de participación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación
CURRICULUM	Y	BLOB	Corresponde al currículum en digital

Tabla B.36. Atributos de la tabla MGI_TAB_INVESTIGADORES.

Fuente: [Propia].

➤ **MGI_TAB_LINEABASE**

Entidad que almacena información de las líneas base del marco lógico que tiene el proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
MARCO_LOG_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del marco lógico
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.37. Atributos de la tabla MGI_TAB_LINEA_BASE.

Fuente: [Propia].

➤ **MGI_TAB_LINEA_INVE_PROYECTO**

Entidad que almacena información de las líneas de investigación del proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
POLITICA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la política de investigación
AREA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
LINEA_INV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la línea de investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.38. Atributos de la tabla MGI_TAB_LINEA_INVE_PROYECTO.

Fuente: [Propia].

➤ **MGI_TAB_LINEA_INVESTIGACION**

Entidad que almacena información de las líneas de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (300 Byte)	Corresponde a la descripción
AREA_INV_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del área de investigación
OBSERVACION	Y	VARCHAR2 (300 Byte)	Corresponde a la observación

Tabla B.39. Atributos de la tabla MGI_TAB_LINEA_INVESTIGACION.

Fuente: [Propia].

➤ **MGI_TAB_MARCO_LOGICO**

Entidad que almacena información del marco lógico del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
FUNCION	Y	VARCHAR2 (1 Byte)	Corresponde a la función
COSTO	Y	NUMBER	Corresponde al costo
COSTO_CONTRAPARTE	Y	NUMBER	Corresponde al costo de la contraparte
FECHA_INICIO	Y	DATE	Corresponde a la fecha de inicio
FECHA_FIN	Y	DATE	Corresponde a la fecha de finalización
RESPONSABLE	Y	VARCHAR2 (200 Byte)	Corresponde al responsable
SECUENCIA1	Y	NUMBER	Corresponde a la secuencia uno
SECUENCIA2	Y	NUMBER	Corresponde a la secuencia dos
SECUENCIA3	Y	NUMBER	Corresponde a la secuencia tres
SECUENCIA4	Y	NUMBER	Corresponde a la secuencia cuatro
SECUENCIA5	Y	NUMBER	Corresponde a la secuencia cinco
NUMERACION	Y	VARCHAR2 (10 Byte)	Corresponde a la numeración
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
PADRE_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código padre
TIPO_OBJ_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al tipo de objetivo
OBSERVACION	Y	VARCHAR2 (50 Byte)	Corresponde a la observación
ACTIVIDAD_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la actividad

Tabla B.40. Atributos de la tabla MGI_TAB_MARCO_LOGICO.

Fuente: [Propia].

➤ MGI_TAB_MES_PERIODO

Entidad que almacena información de los costos del proyecto por periodos, los mismos que se descomponen en meses.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
MONTO	Y	NUMBER	Corresponde al monto requerido
GASTADO	Y	NUMBER	Corresponde al monto gastado
SALDO	Y	NUMBER	Corresponde al saldo disponible
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
PERIODO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del período
MES_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del mes
NUMERO	Y	NUMBER	Corresponde al número

Tabla B.41. Atributos de la tabla MGI_TAB_MES_PERIODO.

Fuente: [Propia].

➤ MGI_TAB_NIVEL_PARTICIPACION

Entidad que almacena información de los tipos de niveles de participación en los que una institución asociada al proyecto de investigación puede estar.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.42. Atributos de la tabla MGI_TAB_NIVEL_PARTICIPACION.

Fuente: [Propia].

➤ MGI_TAB_OBSERVACIONES

Entidad que almacena información de las observaciones realizadas al proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
FECHA	Y	DATE	Corresponde a la fecha de observación
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde al comentario
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
EVALUADOR_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del evaluador

Tabla B.43. Atributos de la tabla MGI_TAB_OBSERVACIONES.

Fuente: [Propia].

➤ MGI_TAB_PARAMETROS_EVALUACION

Entidad que almacena información de los parámetros con los que se evaluarán los proyectos, programas y proyectos de aula.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
VALOR_MAXIMO	Y	NUMBER (9,2)	Corresponde al valor máximo
DEFI_INVE_CODIGO	N	VARCHAR2 (10 Byte)	Código definición de la investigación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.44. Atributos de la tabla MGI_TAB_PARAMETROS_EVALUACION.

Fuente: [Propia].

➤ MGI_TAB_PARAMETROS_PROYECTO

Entidad que almacena información de los parámetros del proyecto, que a la vez contienen a los parámetros de evaluación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
VALOR_MAXIMO	Y	NUMBER (9,2)	Corresponde al valor máximo
PARAMETRO_EVAL_CODIGO	N	VARCHAR2 (10 Byte)	Código parámetro evaluación
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.45. Atributos de la tabla MGI_TAB_PARAMETROS_PROYECTO.

Fuente: [Propia].

➤ **MGI_TAB_PATENTES**

Entidad que almacena información de las patentes de los artículos científicos publicados como resultado de la ejecución de un proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
TIPO_PATENTE	Y	VARCHAR2 (20 Byte)	Corresponde al tipo de patente
LUGAR_REGISTRO	Y	VARCHAR2 (50 Byte)	Corresponde al lugar de registro
NUMERO_REGISTRO	Y	NUMBER	Corresponde al número de registro
FECHA_REGISTRO	Y	DATE	Corresponde a la fecha de registro
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto

Tabla B.46. Atributos de la tabla MGI_TAB_PATENTES.

Fuente: [Propia].

➤ **MGI_TAB_PERFIL_PROYECTO**

Entidad que almacena información del perfil del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
CONTENIDO	Y	LONG	Corresponde al contenido
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
ESTRUCTURA_FORM_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la estructura del formato
VIGENCIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de vigencia del formato
TIPO_FORM_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del tipo de formato
OBSERVACION	Y	VARCHAR2 (200 Byte)	Corresponde a la observación

Tabla B.47. Atributos de la tabla MGI_TAB_PERFIL_PROYECTO.

Fuente: [Propia].

➤ **MGI_TAB_PERIODO**

Entidad que almacena información de los periodos.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.48. Atributos de la tabla MGI_TAB_PERIODO.

Fuente: [Propia].

➤ **MGI_TAB_PERIODO_PROYECTO**

Entidad que almacena información de los períodos de duración los proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ANIO	Y	NUMBER (4)	Corresponde al año
DESDE	Y	DATE	Corresponde a la fecha de inicio
HASTA	Y	DATE	Corresponde a la fecha de finalización
MONTO	Y	NUMBER	Corresponde al monto requerido
GASTADO	Y	NUMBER	Corresponde al valor gastado
SALDO	Y	NUMBER	Corresponde al saldo
PERIODO_CERRADO	Y	VARCHAR2 (1 Byte)	Corresponde al período cerrado
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
PERIODO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del período

Tabla B.49. Atributos de la tabla MGI_TAB_PERIODO_PROYECTO.

Fuente: [Propia].

➤ MGI_TAB_POL_INVESTIGACION

Entidad que almacena información de las políticas de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (300 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (300 Byte)	Corresponde a la observación

Tabla B.50. Atributos de la tabla MGI_TAB_POL_INVESTIGACION.

Fuente: [Propia].

➤ MGI_TAB_PORCENTAJE_EVALUACION

Entidad que almacena información de los porcentajes de evaluación que sirven para controlar el porcentaje que deben sumar los proyectos de investigación una vez que estos hayan sido evaluados.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
MAXIMO	Y	NUMBER	Corresponde al valor máximo
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación
DEFI_INVE_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde a la definición de la investigación
MINIMO	Y	NUMBER	Corresponde al valor mínimo

Tabla B.51. Atributos de la tabla MGI_TAB_PORCENTAJE_EVALUACION.

Fuente: [Propia].

➤ MGI_TAB_PROBLEMAS

Entidad que almacena información de los problemas encontrados durante la ejecución del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
PROBLEMA	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción del problema

SOLUCION	Y	VARCHAR2 (100 Byte)	Corresponde a la solución empleada
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe

Tabla B.52. Atributos de la tabla MGI_TAB_PROBLEMAS.

Fuente: [Propia].

➤ MGI_TAB_PROCEDIMIENTO_CONV

Entidad que almacena información del procedimiento y condiciones que los investigadores deben cumplir para la presentación de proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (4000 Byte)	Corresponde a la descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
OBSERVACION	Y	VARCHAR2 (300 Byte)	Corresponde a la observación

Tabla B.53. Atributos de la tabla MGI_TAB_PROCEDIMIENTO_CONV.

Fuente: [Propia].

➤ MGI_TAB_PRODUCION_CIENTIFICA

Entidad que almacena información de la producción científica del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (20 Byte)	Corresponde al código
TITULO	Y	VARCHAR2 (100 Byte)	Corresponde al título
TIPO	Y	VARCHAR2 (20 Byte)	Corresponde al tipo
FECHA_INICIO	Y	DATE	Corresponde a la fecha de inicio
FECHA_TERMINACION	Y	DATE	Corresponde a la fecha de terminación
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe

Tabla B.54. Atributos de la tabla MGI_TAB_PRODUCION_CIENTIFICA.

Fuente: [Propia].

➤ MGI_TAB_PROMEDIO_PARAMETRO

Entidad que almacena el promedio de los proyectos de investigación por cada uno de sus parámetros.

Atributo	¿Nulo?	Tipo Dato	Descripción
NUMERO_NOTAS	Y	NUMBER	Corresponde al número de notas
SUMA_TOTAL	Y	NUMBER	Corresponde a la suma total por proyecto
PROMEDIO	Y	NUMBER	Corresponde al promedio

CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
PARAMETRO_PROY_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del parámetro

Tabla B.55. Atributos de la tabla MGI_TAB_PROMEDIO_PARÁMETRO.

Fuente: [Propia].

➤ MGI_TAB_PROYECTO_EVALUADOR

Entidad que almacena información de los proyectos asignados a los evaluadores para su respectiva evaluación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
DESCRIPCION	Y	VARCHAR2 (100 Byte)	Corresponde a la descripción
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
EVALUADOR_CODIGO	N	VARCHAR2 (20 Byte)	Corresponde al código del evaluador

Tabla B.56. Atributos de la tabla MGI_TAB_PROYECTO_EVALUADOR.

Fuente: [Propia].

➤ MGI_TAB_PROYECTOS

Entidad que almacena información de los proyectos presentados, los cuales pasan por una serie de etapas hasta la culminación de estos.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
TITULO_CORTO	N	VARCHAR2 (100 Byte)	Corresponde al título corto
TITULO_COMPLETO	N	VARCHAR2 (2000 Byte)	Corresponde al título completo
INSTITUCION_EJECUTORA	N	VARCHAR2 (100 Byte)	Corresponde a la institución ejecutora
DURACION	N	NUMBER	Corresponde a la duración
FECHA_INICIO	N	DATE	Corresponde a la fecha de inicio
FECHA_FIN	N	DATE	Corresponde a la fecha de finalización
FECHA_REGISTRO	N	DATE	Corresponde a la fecha de registro
MONTO_PROYECTO	Y	NUMBER	Corresponde al monto requerido
PRESUPUESTADO	Y	NUMBER	Corresponde al monto

			presupuestado
EJECUTADO	Y	NUMBER	Corresponde al monto ejecutado
SALDO	Y	NUMBER	Corresponde al monto disponible
ASIGNACION_CERRADA	Y	VARCHAR2 (1 Byte)	Corresponde a la asignación cerrada
PRIORIDAD	Y	NUMBER	Corresponde a la prioridad
PROBLEMA	Y	VARCHAR2 (4000 Byte)	Corresponde al problema
JUSTIFICACION	Y	VARCHAR2 (4000 Byte)	Corresponde a la justificación
INVESTIGADOR_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
DOCENTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del docente
EMPLEADO_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del empleado
ESTUDIANTE_CEDULA	Y	VARCHAR2 (20 Byte)	Corresponde a la cédula del estudiante
CONVOCATORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la convocatoria
TIPO_CONV_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al tipo de convocatoria
DEPEN_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la dependencia
INST_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la institución
DEFI_INVE_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la definición de la investigación
ESTADO_PRO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del estado del proyecto
TIPOLOGIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la tipología
LOCALIDAD_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la localidad
CATEGORIA_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la categoría
TIPO_LOCALIDAD_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del tipo de localidad
PROYINST_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al

			código del proyecto institucional
OBSERVACION	N	VARCHAR2 (4000 Byte)	Corresponde a la observación
TIPO_PROY_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del tipo de proyecto
DRSOLUCION_TSESION_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde a código del tipo de sesión
DRSOLUCION_PTORDEN_NUMERO	Y	NUMBER (5)	Corresponde al número de punto de orden
DRSOLUCION_RESOLUCION_NUMERO	Y	NUMBER (10)	Corresponde al número de resolución
DRSOLUCPERCONSEJO_FECHA_INICIO	Y	DATE	Corresponde a la fecha de inicio del consejo directivo
DRSOLUCION_SESION_FECHA	Y	DATE	Corresponde a la fecha de la sesión
DRSOLUCION_CONSEJO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del consejo

Tabla B.57. Atributos de la tabla MGI_TAB_PROYECTOS.

Fuente: [Propia].

➤ MGI_TAB_PROYECTOS_RELACIONADOS

Entidad que almacena información de los proyectos relacionados.

Atributo	¿Nulo?	Tipo Dato	Descripción
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
PROYREL_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto relacionado
TITULO	Y	VARCHAR2 (2000 Byte)	Corresponde al título
ESTADO	Y	VARCHAR2 (50 Byte)	Corresponde al estado
INSTITUCION_EJECUTORA	Y	VARCHAR2 (100 Byte)	Corresponde a la institución ejecutora
DIRECTOR	Y	VARCHAR2 (100 Byte)	Corresponde al director
DIRECCION	Y	VARCHAR2 (100 Byte)	Corresponde a la dirección
TELEFONO	Y	VARCHAR2 (10 Byte)	Corresponde al teléfono
EMAIL	Y	VARCHAR2 (50 Byte)	Corresponde al email
FECHA_INICIO	Y	DATE	Corresponde a la fecha de inicio
FECHA_FIN	Y	DATE	Corresponde a la fecha de finalización
EJECUTADO_POR	Y	VARCHAR2 (10 Byte)	Corresponde al nombre de quien lo ejecutó

Tabla B.58. Atributos de la tabla MGI_TAB_PROYECTOS.

Fuente: [Propia].

➤ **MGI_TAB_PUBLICACIONES**

Entidad que almacena información de las publicaciones realizadas por parte de los investigadores, como parte de su currículum.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
AUTORES	Y	VARCHAR2 (200 Byte)	Corresponde a los autores
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde a la descripción
ANIO	Y	VARCHAR2 (10 Byte)	Corresponde al año
INSTITUCION	Y	VARCHAR2 (200 Byte)	Corresponde a la institución
LUGAR	Y	VARCHAR2 (100 Byte)	Corresponde al lugar
PERSONA_CEDULA	N	VARCHAR2 (20 Byte)	Corresponde a la cédula del investigador
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.59. Atributos de la tabla MGI_TAB_PUBLICACIONES.

Fuente: [Propia].

➤ **MGI_TAB_RESULTADOS**

Entidad que almacena información de los resultados obtenidos con el desarrollo del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde a la descripción
USUARIO_BENEFICIARIO	Y	VARCHAR2 (50 Byte)	Corresponde a los beneficiarios
COMENTARIO	Y	VARCHAR2 (200 Byte)	Corresponde al comentario
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del informe

Tabla B.60. Atributos de la tabla MGI_TAB_RESULTADOS.

Fuente: [Propia].

➤ **MGI_TAB_RUBRO_MES**

Entidad que almacena los rubros mensuales que demanda el proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
MONTO	Y	NUMBER	Corresponde al monto asignado
GASTADO	Y	NUMBER	Corresponde al monto gastado
SALDO	Y	NUMBER	Corresponde al monto disponible
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
PERIODO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del período
MES_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del mes
TIPO_RUBRO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de tipo de rubro

Tabla B.61. Atributos de la tabla MGI_TAB_RUBRO_MES.

Fuente: [Propia].

➤ **MGI_TAB_RUBRO_PERIODO**

Entidad que almacena información de los rubros que demanda el proyecto por período.

Atributo	¿Nulo?	Tipo Dato	Descripción
NUMERO	Y	NUMBER	Corresponde al código
INGRESOS	Y	NUMBER	Corresponde a los ingresos
MONTO	Y	NUMBER	Corresponde al monto asignado
GASTADO	Y	NUMBER	Corresponde al monto gastado
SALDO	Y	NUMBER	Corresponde al monto disponible
PERIODO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del período
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
TIPO_RUBRO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de tipo de rubro
ESTE_MES	Y	NUMBER	Corresponde al monto del mes

Tabla B.62. Atributos de la tabla MGI_TAB_RUBRO_PERIODO.

Fuente: [Propia].

➤ **MGI_TAB_RUBRO_PERIODO_DET**

Entidad que almacena información del detalle de los gastos que demanda el proyecto de investigación por período.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
NOMBRE	Y	VARCHAR2 (200 Byte)	Corresponde al nombre
DESCRIPCION	Y	VARCHAR2 (200 Byte)	Corresponde a la descripción
CANTIDAD	Y	NUMBER	Corresponde a la cantidad
COSTO_UNITARIO	Y	NUMBER	Corresponde al costo unitario
PERIODO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del período
PROYECTO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
TIPO_RUBRO_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código del tipo de rubro
UNIDAD_MED_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la unidad de medida

Tabla B.63. Atributos de la tabla MGI_TAB_RUBRO_PERIODO_DET.

Fuente: [Propia].

➤ **MGI_TAB_SECT_IMPA_PROYECTOS**

Entidad que almacena información de los sectores en los que tendrá impacto el proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
SECIMP_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del sector de impacto
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.64. Atributos de la tabla MGI_TAB_SECT_IMPA_PROYECTOS.

Fuente: [Propia].

➤ **MGI_TAB_SECTOR_IMPACTO**

Entidad que almacena información de las diferentes definiciones de sectores de impacto.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde a la descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.65. Atributos de la tabla MGI_TAB_SECTOR_IMPACTO.

Fuente: [Propia].

➤ **MGI_TAB_SUPUESTOS**

Entidad que almacena información de las suposiciones del marco lógico.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
MARCO_LOG_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del marco lógico
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.66. Atributos de la tabla MGI_TAB_SUPUESTOS.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_CONVOCATORIA**

Entidad que almacena información de los tipos de convocatoria.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.67. Atributos de la tabla MGI_TAB_TIPO_CONVOCATORIA.

Fuente: [Propia]

➤ **MGI_TAB_TIPO_DOCUMENTO_RESP**

Entidad que almacena información de los tipos de documentos de respaldo.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.68. Atributos de la tabla MGI_TAB_TIPO_DOCUMENTO_RESP.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_EVENTO**

Entidad que almacena información de los tipos de evento.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.69. Atributos de la tabla MGI_TAB_TIPO_EVENTO.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_FORMATO**

Entidad que almacena información de los tipos de formato.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
SIGLAS	Y	VARCHAR2 (50 Byte)	Corresponde a las siglas
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.70. Atributos de la tabla MGI_TAB_TIPO_FORMATO.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_INFORME**

Entidad que almacena información de los tipos de informe.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.71. Atributos de la tabla MGI_TAB_TIPO_INFORME.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_LOCALIDAD**

Entidad que almacena información de los tipos de localidad.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.72. Atributos de la tabla MGI_TAB_TIPO_LOCALIDAD.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_OBJETIVO**

Entidad que almacena información de los tipos de objetivo.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.73. Atributos de la tabla MGI_TAB_TIPO_OBJETIVO.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_PARTICIPACION**

Entidad que almacena información de los tipos de participación.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.74. Atributos de la tabla MGI_TAB_TIPO_PARTICIPACION.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_PROYECTO.**

Entidad que almacena información de los tipos de proyecto.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.75. Atributos de la tabla MGI_TAB_TIPO_PROYECTO.

Fuente: [Propia].

➤ **MGI_TAB_TIPO_RUBRO**

Entidad que almacena información de los tipos de rubro.

Atributo	¿Nulo?	Tipo Dato	Descripción
ESTADO	Y	VARCHAR2 (1 Byte)	Corresponde al estado
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
TIPO_FORM_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del tipo de formato
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.76. Atributos de la tabla MGI_TAB_TIPO_RUBRO.

Fuente: [Propia].

➤ **MGI_TAB_TIPOLOGIA**

Entidad que almacena información de la tipología del proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
DESCRIPCION	Y	VARCHAR2 (50 Byte)	Corresponde al descripción
OBSERVACION	Y	VARCHAR2 (100 Byte)	Corresponde a la observación

Tabla B.77. Atributos de la tabla MGI_TAB_TIPOLOGIA.

Fuente: [Propia].

➤ **MGI_TAB_TRANSFERENCEIA**

Entidad que almacena información de la transferencia tecnológica realizada como resultado de la ejecución de un proyecto de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
ORGANISMO_BENEFICIARIO	Y	VARCHAR2 (100 Byte)	Corresponde al organismo beneficiario
TECNOLOGIA_TRANSFERIDA	Y	VARCHAR2 (100 Byte)	Corresponde la tecnología transferida
METODO_TRANSFERENCEIA	Y	VARCHAR2 (50 Byte)	Corresponde al método de transferencia
PROPOSITO	Y	VARCHAR2 (100 Byte)	Corresponde al propósito
RESULTADO	Y	VARCHAR2 (100 Byte)	Corresponde al resultado
PROYECTO_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del proyecto
INFORME_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde código informe

Tabla B.78. Atributos de la tabla MGI_TAB TRASFERENCEIA.

Fuente: [Propia].

➤ **MGI_TAB_USUARIOS**

Entidad que almacena información de las cuentas de usuario del sistema de gestión de la investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
OBSERVACION	Y	VARCHAR2 (300 Byte)	Corresponde a la observación
DEPEN_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la dependencia
INST_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código de la institución
USUARIOS_CUENTA	N	VARCHAR2 (100 Byte)	Corresponde a la cuenta de usuario

Tabla B.79. Atributos de la tabla MGI_TAB_USUARIOS.

Fuente: [Propia].

➤ **MGI_TAB_VIGENCIA_FORMATO**

Entidad que almacena información de la fecha de vigencia que tienen los formatos para la presentación de proyectos de investigación.

Atributo	¿Nulo?	Tipo Dato	Descripción
VIGENTE	Y	VARCHAR2 (1 Byte)	Corresponde a si está vigente o no
CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código
FECHA_VIGENCIA	N	DATE	Corresponde a la fecha de vigencia
DESCRIPCION	Y	VARCHAR2 (500 Byte)	Corresponde a la descripción
TIPO_FORM_CODIGO	N	VARCHAR2 (10 Byte)	Corresponde al código del tipo de formato
DEFI_INVE_CODIGO	Y	VARCHAR2 (10 Byte)	Corresponde al código de la definición de la investigación

Tabla B.80. Atributos de la tabla MGI_TAB_VIGENCIA_FORMATO.

Fuente: [Propia].

C. GUÍA DE PROGRAMACIÓN

C.1. ESTÁNDARES DE PROGRAMACIÓN

Un aspecto muy importante al emprender un proyecto informático es la estandarización de normas y políticas que permitan la mejor comprensión de los documentos, código de programación, implementación de la base de datos y demás recursos inmersos, entre las personas relacionadas en el desarrollo.

Se ha comprobado que las personas encargadas del mantenimiento de la aplicación pasan la mitad del tiempo tratando de interpretar bloques de código implementado por terceras erróneamente escritas, es por esta razón que este documento pretende dar los lineamientos necesarios que permitan un mejor entendimiento de la codificación implementada en el diseño y desarrollo del “Sistema de Gestión de la investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN”.

➤ Propósito

El presente documento tiene como finalidad dar a conocer a los interesados los estándares de programación que rigen en el desarrollo y mantenimiento de la aplicación implementada, el mismo que servirá de base para las aplicaciones futuras.

➤ Descripción

El presente documento presenta al interesado las reglas y políticas que permita estandarizar y normar el desarrollo del “Sistema de Gestión de la Investigación Científica, Tecnológica e Innovativa para el Centro Universitario de Investigación Científica y Tecnológica de la UTN” utilizando la plataforma Oracle ® 10g como servidor de base de datos, OAS (Oracle® Application Server 10.1.2) como servidor de aplicaciones y como IDE de programación Oracle® Developer Suite 10.1.2 con lenguaje de programación PL/SQL.

Para una mejor comprensión del presente documento, la persona interesada deberá tener los conocimientos necesarios sobre las siguientes tecnologías:

- Bases de Datos;
- Conocimientos sobre la tecnología Oracle ®;
- Lenguaje de programación PL/SQL (4gl de Oracle ®).

C.2. ESTANDARIZACIÓN DEL DISEÑO DE LA BASE DE DATOS

En las siguientes líneas se desea implementar las pautas que normalicen el diseño e implementación del repositorio de información o base de datos.

➤ Objetos de una Base de Datos en Oracle ® 10g

Los siguientes son tipos de objetos que se pueden definir en una BDD Oracle ®10g.

Objeto	Prefijo
Tabla	TAB
Trigger	TRG
Storeprocedure (Procedimientos almacenados)	PRO
Views (Vistas)	VIW
Sequence (Secuencias)	SEQ
Synonym (Sinónimos)	SYN
Function(Funciones)	FUN
Index	IDX
Foreingkey (claves foráneas)	FK
Primarykey (Claves primarias)	PK
Unique (Claves Únicas)	UK
Constraints Not Null	NN
Paquete	PKG
Constraints Check	CK

Tabla C.1. Tipos de Objetos de la Base de Datos.

Fuente: [Propia].

➤ Áreas de Desarrollo

Las áreas de desarrollo de la Universidad Técnica del Norte están clasificadas de la siguiente forma:

Área	Prefijo
Académico	ACA
Recaudación	REC
Recursos Humanos	RHU
Resoluciones Órganos Colegiados (consejos y comisiones UTN)	RHC
Nómina	NOM
Inventarios	INV
Adquisiciones	ADQ
Bienestar Socioeconómico	BSE
Bienestar Salud	BIS
Bienestar Odontológico	BIO
Bienestar Laboratorio Clínico	BIC
Seguridad	SEG
Auditoría	AUD

Tabla C.2. Áreas de Desarrollo.

Fuente [Propia].

➤ Nombres de los Objetos de la Base de Datos

Una vez definidos los tipos de objetos que maneja Oracle ® y las áreas de desarrollo de módulos en La Institución, se sugiere crear los objetos de la base de datos utilizando la siguiente nomenclatura:

El nombre del objeto empezará por el prefijo del área de desarrollo, seguido por el prefijo del tipo de objeto, seguido del símbolo _ más el nombre del objeto que represente a la entidad.

A continuación se presenta algunos ejemplos de nombres de objetos:

Área de Desarrollo	Tipo de Objeto	Nombre del Objeto
Académico	Tabla	ACA_TAB_ESTUDIANTES
Uso Común	Tabla	UCO_TAB_PERSONA
Investigación	Procedure	MGI_PRO_ADELANTAR_FECHA
Uso Común	Primary Key	UCO_PK_CEDULA

Tabla C.3. Nombres de los Objetos.

Fuente: [Propia].

Si los nombres de los objetos son compuestos, como por ejemplo en la definición de nombres de claves foráneas, que relacionan 2 tablas, se intercalará el símbolo _ entre cada nombre de la tabla, por ejemplo:

Área de Desarrollo	Tipo de Objeto	Nombre del Objeto
Académico	Foreign Key	ACA_FK_ESTUDIANTE_MATRICULA
Uso Común	Foreign Key	UCO_FK_SEXO_PERSONA
Recursos Humanos	Foreign Key	RHU_FK_EMPLEADO_TITULOS

Tabla C.4. Ejemplos de Definición de Objetos.

Fuente: [Propia].

C.3. ESTANDARIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN

Las políticas y normas que rigen la codificación de los procesos en el lenguaje de programación elegido es la siguiente:

➤ Nombres de Objetos

Para la declaración de variables de objetos como formularios, cuadros de texto, checkbox y otros se seguirá la siguiente nomenclatura:

Los nombres de los objetos empezarán con el prefijo de 3 caracteres que corresponde al prefijo relacionado con el tipo de objeto, seguido del nombre que se quiere asignar a la variable. Por ejemplo: CMD_ACEPTAR.

En el caso de que el nombre de la variable conste de 2 o más palabras el nombre del objeto empezará con un prefijo de 3 caracteres correspondiente al prefijo del objeto, seguido por el

carácter _ más el nombre del que se desea asignar al objeto, en cada palabra se intercalara el carácter _ para una mejor comprensión.

Por ejemplo: CMD_GENERAR_LISTA_ASISTENTES.

A continuación se listan los prefijos de objetos más comunes que se utilizan en algunos de los lenguajes de programación más utilizados.

Objeto	Prefijo	Ejemplo
Alertas	ALE	ALE_AVISO
Botones (CommandButton)	CMD	CMD_ACEPTAR
Checkbox	CHK	CHK_SEXO
Cuadros combinados (ComboBox)	LOV	LOV_ESTUDIANTES
Formulario (Form)	FRM	FRM_PRINCIPAL
Lista de Valores	LOV	LOV_ESTUDIANTES
Cuadros de Texto (TextBox)	TXT	TXT_NOMBRE
Etiquetas (Label)	LBL	LBL_NOMBRE
Botones de Radio (RadioButton)	RDB	RDB_TIPO_FACTURA
Windows	WIN	WIN_CLIENTES
Canvas o lienzos	CAN	CAN_CLIENTES
Grupos de Registros	GRE	GRE_CLIENTES

Tabla C.5. Prefijos de Objetos.
Fuente: [Propia].

➤ Nombres de Variables de Tipos de Datos

Para el uso de tipos de datos primitivos se sugiere utilizar la siguiente nomenclatura:

El nombre de la variable de tipo primitivo empezara señalando el ámbito de acción: l(local), g(global), p(parámetro), luego un prefijo de 3 dígitos que identifique el tipo de dato que va a almacenar la variable seguido con el nombre de la variable cuya primera letra empieza con mayúscula, si el nombre de la variable está compuesta por 2 o más palabras se intercalara la letra mayúscula de cada palabra, no existirá espacios en blanco, guiones ni subrayados en la unión de las palabras. Ejemplos: intValor, intValorCoordenada, douValorIva.

En la *Tabla B.6.* Se muestra el prefijo asociado al tipo de datos que se desea utilizar:

Tipo de dato	Descripción	Prefijo	Ejemplo
Binary_Integer	Antigua versión del PLS_INTEGER. Almacena valores en un rango de - 2147483647 .. 2147483647	bint	lbintSumaTotal, gbintSuma
Bfile	Almacena datos binarios no estructurados en archivos del sistema operativo, fuera de la base de datos. Una columna BFILE almacena un localizador del archivo a uno externo que contiene los datos. Admiten hasta	bfil	gbfilVariable, gfilVariableDePrueba

	4Gigabytes. El administrador de la base de datos debe asegurarse de que exista el archivo en disco y de que los procesos de Oracle ® tengan permisos de lectura para el archivo.		
Blob	Permite almacenar datos binarios no estructurados. Admiten hasta 4 Gigabytes	blo	pbloImagen, gbloImagen
Boolean	Permite almacenar 2 valores lógicos, TRUE O FALSE	boo	gbooRespuesta, pbooRespuesta
Char	Cadena de caracteres (alfanuméricos) de longitud fija. Mínimo 1 y máximo 32767	ch	lchSexo, pchSexo
Clob	Almacena datos de tipo carácter. Admiten hasta 4Gigabytes	clo	gcloArchivo, lcloArchivo
Date	Almacena un punto en el tiempo (fecha y hora). El tipo de datos DATE almacena el año (incluyendo el siglo), el mes, el día, las horas, los minutos y los segundos (después de medianoche). Oracle ® utiliza su propio formato interno para almacenar fechas. Los tipos de datos DATE se almacenan en campos de longitud fija de siete octetos cada uno, correspondiendo al siglo, año, mes, día, hora, minuto, y al segundo. Para entrada/salida de fechas, Oracle ® utiliza por defecto el formato DD-MMM-AA. Para cambiar este formato de fecha por defecto se utiliza el parámetro NLS_DATE_FORMAT. Para insertar fechas que no estén en el mismo formato de fecha estándar de Oracle ®, se puede utilizar la función TO_DATE con una máscara del formato: TO_DATE (el “13 de noviembre de 1992”, “DD del MES, YYYY”)	dat	pdatFechaActual, gdatFechaActual
Dec	Permite declarar números de punto fijo. Máximo precisión de 38 dígitos decimales	dec	ldecTotal, gdecTotal
Decimal	Permite declarar números de punto fijo. Máximo precisión de 38 dígitos decimales	deci	gdeciVariable, gdeciVariableDePrueba
Doubleprecisión	Permite declarar números de punto flotante. Máxima precisión de 126 dígitos binarios, de los cuales 38 son dígitos decimales	doup	ldoupCantidad, gdoupCantidadTemporal
Int	Permite declarar enteros. Precisión máxima de 38 dígitos decimales	int	lintSuma, gintSumaTotal
Integer	Permite declarar enteros. Precisión máxima de 38 dígitos decimales	intg	lintgVariable, pintgSumaValores
Intervalyeararto	Permite manipular intervalos de	intv	pintvFechaCompra,

month	tiempo de años y meses		gintvFechaRegistro
Lob	Permiten almacenar y manipular bloques grandes de datos no estructurados (tales como texto, imágenes, videos, sonidos, etc) en formato binario o del carácter. Admiten hasta 4 Gigabytes. Una tabla puede contener varias columnas de tipo LOB. Soportan acceso aleatorio. Las tablas con columnas de tipo LOB no pueden ser replicadas.	lob	plobVideoAcademico, globTextoUtn
Long	Cadena de caracteres de longitud variable. Como máximo admite hasta 2 GB (2000 MB). Los datos LONG deberán ser convertidos apropiadamente al moverse entre diversos sistemas. Este tipo de datos está obsoleto (en desuso), en su lugar se utilizan los datos de tipo LOB (CLOB, NCLOB). Oracle ® recomienda que se convierta el tipo de datos LONG a alguno LOB si aún se está utilizando. No se puede utilizar en cláusulas WHERE, GROUP BY, ORDER BY, CONNECT BY ni DISTINCT. Una tabla sólo puede contener una columna de tipo LONG. Sólo soporta acceso secuencial.	lon	glonValor; plonTemporal
Long raw	Almacenan cadenas binarias de ancho variable. Hasta 2 GB. En desuso, se sustituye por los tipos LOB.	lonr	llonrVariableDePrueba, plonrVariableTemporal
Float	Almacena tipos de datos numéricos en punto flotante. Es un tipo NUMBER que sólo almacena números en punto flotante	flo	gfloValorIce, pfloIva
Natural	Permite restringir una variable entera a valores solo positivos	nat	gnatValor, pintValorTemporal
Nchar	Cadena de caracteres de longitud fija que sólo almacena caracteres Unicode.	nch	gnchCadena, lnchCadenaTemporal
Nclob	Almacena datos de tipo carácter. Admiten hasta 4Gigabytes. Guarda los datos según el juego de caracteres Unicode nacional.	nclo	
Naturaln	Permite restringir una variable entera a valores solo positivos	natn	pnatnValores, lnatnSumaTotal
Numeric	Permite declarar variables de punto fijo. Máximo precisión de 38 dígitos decimales	num	
Number	Almacena números fijos y en punto flotantes. Se admiten hasta 38 dígitos de precisión y son portables a cualquier entre los diversos sistemas en que funcione Oracle ®. Para declarar un tipo de datos NUMBER en	numb	numSumaTotales, pnumSuma

	<p>un CREATE ó UPDATE es suficiente con: nombre_columna NUMBER</p> <p>opcionalmente se le puede indicar la precisión (número total de dígitos) y la escala (número de dígitos a la derecha de la coma, decimales, los cogerá de la precisión indicada): nombre_columna NUMBER (precision, escala)</p> <p>Si no se indica la precisión se tomará en función del número a guardar, si no se indica la escala se tomará escala cero. Para no indicar la precisión y sí la escala podemos utilizar: nombre_columna NUMBER (*, escala)</p> <p>Para introducir números que no estén el formato estándar de Oracle ® se puede utilizar la función TO_NUMBER.</p>		
Nvarchar2	Cadena de caracteres de longitud variable que sólo almacena caracteres Unicode. Mínimo 1 y máximo 32767	nvar	pnvarCadena, gnvarCadenaDatos
Pls_integer	Tiene la misma funcionalidad que el number, pero ocupa menos espacio y mejor desempeño. El rango de magnitud está entre -2147483647 y 2147483647	pls	pplsValorPrueba, gplsValorTotal
Positive	Permite restringir una variable entera a valores solo positivos	pos	pposTemporal, gposSumaTemporal
Positiven	Permite restringir una variable entera a valores solo positivos, que no admiten valores nulos.	posn	lposnCoordenada, gposnSumaTotal
Raw	Almacenan cadenas binarias de ancho variable. Hasta 32767 bytes. En desuso, se sustituye por los tipos LOB.	raw	prawValorTemporal, growPrueba
Real	Almacena valores con punto flotante. Almacena en un rango de 63 dígitos binarios, 18 para dígitos decimales	rea	preaSumaIva, greaSumaEstudiantes
Row		row	prowVariableDePrueba, growTemporal
Rowid	Almacenar la dirección única de cada fila de la tabla de la base de datos. ROWID físico almacena la dirección de fila en las tablas, las tablas en clúster, los índices, excepto en las índices-organizados (IOT). ROWID lógico almacena la dirección de fila en tablas de índice-organizado (IOT). Un ejemplo del valor de un campo ROWID podría ser: "AAAIugAAJAAC4AhAAI". El formato es el siguiente: Para "OOOOOFFFFBBBBBBRRR", donde: OOOOOO: segmento de la base de datos (AAAIug en el ejemplo).	rowid	prowidVariableTempora l, growidPrueba

	<p>Todos los objetos que estén en el mismo esquema y en el mismo segmento tendrán el mismo valor.</p> <p>FFF: el número de fichero del tablespace relativo que contiene la fila (fichero AAJ en el ejemplo).</p> <p>BBBBBB: el bloque de datos que contiene a la fila (bloque AAC4Ah en el ejemplo). El número de bloque es relativo a su fichero de datos, no al tablespace. Por lo tanto, dos filas con números de bloque iguales podrían residir en diferentes datafiles del mismo tablespace.</p> <p>RRR: el número de fila en el bloque (fila AAI en el ejemplo). Este tipo de campo no aparece en los SELECT ni se puede modificar en los UPDATE, ni en los INSERT. Tampoco se puede utilizar en los CREATE. Es un tipo de datos utilizado exclusivamente por Oracle ®. Sólo se puede ver su valor utilizando la palabra reservada ROWID, por ejemplo: <i>selectrowid, nombre, apellidos from clientes</i></p> <p>Ejemplo 2: <i>SELECT ROWID, SUBSTR(ROWID,15,4) "Fichero", SUBSTR(ROWID,1,8) "Bloque", SUBSTR(ROWID,10,4) "Fila" FROM proveedores</i></p> <p>Ejemplo 3: una forma de saber en cuántos ficheros de datos está alojada una tabla: SELECT COUNT(DISTINCT(SUBSTR(ROWID,7,3))) "Numeroficheros " FROM facturacion</p>		
Signtype	Permite restringir los valores de una variable a uno de los estados -1,0 y 1. Restringe una variable a uno de los 3 estados -1,0 y 1	sgt	psgtValorCoordenada, lsgtEstado
Smallint	Máxima precisión de 38 dígitos decimales	smal	gsmalValorGlobal, psmalTemporal
Timestamp	Almacena toda una fecha completa, año, mes, día, hora, minuto, segundo. Los valores aceptados en fracción de segundos van desde 0 a 9, por defecto es 6	tims	ltimsFechaNacimiento, gtimsFechaActual
Timestampwith Timezone	Almacena datos de tipo hora incluyendo la zona horaria (explícita), fraccionando los segundos. Los valores aceptados en fracción de segundos van desde 0 a 9, por defecto es 6.	timswt	ptimswtVariable, ltimswtTemporal
Timestamp	Almacena datos de tipo hora	timswlt	ptimswltFecha,

with Local timezone	incluyendo la zona horaria local (relativa), fraccionando los segundos. Cuando se usa un SELECT para mostrar los datos de este tipo, el valor de la hora será ajustado a la zona horaria de la sesión actual		gtimswltFechaTemporal
Urowid	ROWID universal. Admite ROWID a tablas que no sean de Oracle®, tablas externas. Admite tanto ROWID lógicos como físicos, tiene un tamaño máximo de 4000 bytes	urow	purowVariable, gurowVariableTemporal
Varchar	Cadena de caracteres de longitud variable. Mínimo 1 y máximo 32767	varc	pvarcCadena, lvarcCadenaTemporal
Varchar2	Cadena de caracteres de longitud variable. Mínimo 1 y máximo 32767	varc2	gvarc2Nombre, pvarc2ApellidoEstudiante
XMLType	Tipo de datos abstracto. En realidad se trata de un CLOB. Se asocia a un esquema XML para la definición de su estructura.	xml	pxmlEstructura, lxmlEstructuraDocum

Tabla C.6. Variables y Tipos de Datos.

Fuente: [Propia].

➤ Nombres de constantes

Todo nombre de constante empezará por el prefijo siguiente: C_, seguido del tipo de dato que representa más el nombre de la constante.

Por ejemplo: C_floPorcentajeIva, C_douPi

➤ Nombres de Funciones y procedimientos

Todos los nombres de funciones y procedimientos estarán escritos en MAYUSCULAS, si el nombre de la función o procedimiento está compuesto de 2 o más palabras se lo hará intercalando el signo _ entre cada palabra, por ejemplo:

- FUN_CALCULOS_EDAD_ESTUDIANTES
- PRO_PROCEDIMIENTO
- PKG_PAQUETE.FUN_CALCULO

➤ Documentación y comentarios en el código

Todo bloque de código tendrá como encabezado las siguientes líneas:

/* Creado por: Egresado. Cristian Patricio Herrera Morales

Fecha de creación: 02/09/2012

Última modificación: 07/09/2012

Descripción del bloque: Una breve descripción sobre el bloque de código

Descripción de Variables: Una breve descripción de las variables utilizadas y su utilización en el bloque de código. */

➤ **Palabras Reservadas del Lenguaje de Programación**

Todas las palabras reservadas que forman parte del lenguaje serán escritas en MAYÚSCULA.

Ejemplo:

/* Creado por: Egresado. Cristian Patricio Herrera Morales.

Fecha de creación: 02/09/2012 Última modificación: 07/09/2012

Descripción del bloque: Este bloque permite sumar los subtotaes del campo total

Descripción de Variables:

InumPrueba Variable que almacena la suma de los totales. */

```
PACKAGE BODY CALCULOS IS
```

```
PROCEDURE SUBTOTAL IS
```

```
    InumPrueba NUMERIC;
```

```
BEGIN
```

```
    InumPrueba:=454545;
```

```
    END;
```

```
END;
```

D. MANUAL DE INSTALACIÓN

Instalación de Herramientas Sobre LINUX

Requisitos mínimos de hardware

- Procesador de más de 480 MHz de velocidad.
- Por lo menos 512 Mb de RAM para Linux y 1Gb para Windows.
- El suficiente espacio de disco duro (por lo menos 3Gb).

D.1. ORACLE ® 10G DATABASE SERVER

Se recomienda instalar el motor de base de datos en S.u.S.E. Linux Enterprise Edition versión 9 o 10, incluso soporta la versión 11g sin problemas, pero es necesario incrementar el valor de RAM a por lo menos 1 Gb. Tampoco hubo problemas al instalar el motor de base de datos en Red Hat Enterprise Linux versión 3. Al instalar linux, es necesario configurar el tamaño de la partición swap a por lo menos 1 Gb.

➤ Prerrequisitos

Deben estar instalados los siguientes paquetes en el sistema: glibc, glibc-devel, libstdc++, libstdc++-devel, gcc, gcc-c++, openmotif-libs, openmotif21-libs, pdksh, make, sysstat. Se debe revisar para cada versión de linux cuales son las versiones correctas de los paquetes, se detalla completamente en la Guía de Instalación de Oracle ® 10g Database Server. Para S.u.S.E. Linux se incluye además un paquete de compatibilidad que configura las variables de entorno y scripts en el sistema que es el orarun.

Se debe crear un usuario para poder realizar la instalación, si se quiere instalar en un directorio propio del sistema como /opt o /usr, se le debería dar los permisos necesarios al directorio sobre el cual se instale el producto, aunque no existe ningún problema al instalar en un directorio personalizado que sea propietario el usuario.

Hay que realizar algunos cambios en los parámetros del sistema operativo, se los puede realizar manualmente con el comando sysctl (para obtener información de este comando se puede utilizar el comando “man sysctl” desde la línea de comandos). Los parámetros a cambiar son:

- net.ipv4.ip_local_port_range=1024 65000
- kernel.sem=250 32000 100 128
- kernel.shmmax=2147483648

- fs.file-max=65536

Estos parámetros los incluimos en el archivo /etc/sysctl.conf.

Para S.u.S.E. Linux se debe incluir en el arranque el boot.sysctl de la siguiente manera en la línea de comandos como usuario root:

- chkconfig boot.sysctl //Nos debe dar como resultado boot.sysctl off
- chkconfig boot.sysctl on
- chkconfig boot.sysctl //Nos debe dar como resultado boot.sysctl on

Una vez configurado el boot.sysctl y el archivo /etc/sysctl.conf ejecutamos el comando como root:

- sysctl -p

Y tenemos una salida de los nuevos parámetros configurados.

➤ **Instalación**

Desde el CD de instalación o el directorio en el que se desempaqueto los instaladores ejecutamos el script runInstaller, se ejecuta el Oracle ® Universal Installer, seguimos las instrucciones de acuerdo a las necesidades.

➤ **Después de Instalar**

Se deben configurar algunas variables de entorno (en S.u.S.E. Linux en el .profile y en Red Hat el .bash_profile del usuario del sistema que se definió como administrador de Oracle ®, no el usuario root):

- ORACLE_HOME=<Directorio de Instalación de Oracle >
- ORACLE_SID=<Valor de la Instancia Configurada de Oracle>
- ORACLE_OWNER=<EL usuario del sistema que se definió como Administrador de Oracle>

Son las variables de entorno principales, también se pueden configurar las siguientes:

- NLS_LANG=<Idioma de Oracle, verificar en el Manual de Administración de Oracle>
- CLASSPATH=<Directorio de clases de Java>
- LD_LIBRARY_PATH=<Directorio de librerías binarias de Oracle>
- PATH=<Agregar el Path de los binarios de Oracle que es \$ORACLE_HOME/bin>

➤ **Iniciar y Parar Oracle 10g**

Para iniciar se debe montar las bases de datos y luego subir el listener, complementariamente también el Enterprise Manager de la Base de Datos. Todo esto se lo hace como usuario administrador.

Subir la Base de Datos: desde línea de comandos como usuario administrador ejecutamos.

- `sqlplus /nolog`
- `SQL>connect / as sysdba`
- `SQL>startup`

De igual manera para terminar ejecutamos

- `sqlplus /nolog`
- `SQL> connect / as sysdba`
- `SQL> shutdown [modo de parada abort | immediate | normal | transactional]`
- Cuando la base no está en producción es preferible los modos abort o immediate.

El listener inicia y para con el comando `lsnrctl`:

- `LSNRCTL>start | stop`

El Enterprise Manager inicia o para con el comando:

- `emctl start | stop dbconsole`

D.2. ORACLE ® 10G DEVELOPER SUITE

La instalación no tiene ninguna complicación, se usa los mismos requerimientos que para instalar la base de datos, pero el correcto funcionamiento se da sobre Red Hat Enterprise Edition, igualmente creamos un usuario del sistema para la instalación, pero además agregamos los siguientes paquetes: `compat-glibc`, `compat-libstdc++`, `compat-libstdc++-devel`, `compat-db`, `binutils`, `gnome-libs`, `setarch`. Igualmente las versiones correctas de estos paquetes se encuentran en la Guía de Instalación de Oracle ® Developer Suite 10g.

Desde el CD de instalación o desde los directorios donde se desempaqueto el instalador corremos el script `runInstaller` y seguimos las instrucciones de instalación. Luego tenemos que setear la variable de entorno `ORACLE_HOME` al directorio de instalación.

Para ejecutar los programas, lo hacemos desde el directorio bin donde instalamos, el forms builder es el frmblld.sh y para el reports builder el rwbuilder.sh.

Para correr las formas debemos configurar el mozilla navigator, en el directorio de instalación de mozilla (para Red Hat Enterprise Linux 3 es /usr/lib/mozilla-1.7.10) vamos al directorio de plugins y como usuario root creamos un enlace simbólico a la librería de plugins de java de jdk1.4.2_6 para ns610-gcc32, de esta manera (la librería es libjavaplugin_oji.so):

- `ln -s $ORACLE_HOME/jdk/jre/plugin/ns610-gcc32/libjavaplugin_oji.so`
- `/usr/lib/mozilla-1.7.10/plugins/libjavaplugin_oji.so`
- El mismo procedimiento se utiliza para el mozilla firefox, suele estar instalado en `/usr/lib/firefox`, por lo que el enlace sería:
- `ln -s $ORACLE_HOME/jdk/jre/plugin/ns610-gcc32/libjavaplugin_oji.so`
- `/usr/lib/firefox/plugins/libjavaplugin_oji.so`

Se puede utilizar mozilla, mozilla firefox o netscape navigator, por lo que se realiza el enlace simbólico al directorio plugins donde se encuentre instalado el navegador.

Para iniciar la instancia del contenedor java de aplicaciones para correr las formas ejecutamos el script:

- `$ORACLE_HOME/j2ee/DevSuite/startinst.sh`

Igualmente para detener la instancia usamos el script stopinst.sh. Ya se puede correr las formas en modo de desarrollo.

D.3. ORACLE ® 10G APPLICATION SERVER

De igual manera que en la instalación de los otros paquetes, ejecutamos el script runInstaller desde el disco 1, pero hay que tener otras consideraciones antes de lanzar el script de instalación. Se utilizan los mismos requisitos que para instalar la Oracle ® 10G Database Server, pero además debemos instalar los siguientes paquetes: db1, compat-glibc, compat-libstdc++, compat-libstdc++-devel, compat-db. Se deben desconfigurar las siguientes ENV, ORACLE_HOME, ORACLE_BASE, ORACLE_SID, de la siguiente manera:

- `unset ENV`
- `unset ORACLE_HOME`

- unset ORACLE_BASE
- unset ORACLE_SID

➤ **Instalación**

Una vez hechos los cambios para los requisitos ya por fin se puede lanzar el instalador. Primero se debe instalar la infraestructura, que consiste en Oracle ® LDAP (Oracle ® Internet Directory), para la autenticación del acceso a las aplicaciones, además también se debe escoger la opción Single Sign-on, que sirve para acceso a la web de las aplicaciones Oracle ®, todas estas opciones son parte de Oracle ® Identity Manager, y deben funcionar sobre una instancia especial de Oracle ® 10G Database (Metadatos), preparada especialmente para soportar el Oracle ® Internet Directory, se pueden instalar todo en un solo conjunto o utilizar una instancia ya instalada y prepararla manualmente (lo cual complica las cosas), por lo que es preferible que el instalador realice estas acciones.

Una vez que tenemos instalada la infraestructura, podemos instalar el Oracle ® Application Server, los contenedores OC4J (Oracle ® Application Server Containers for (4) J2EE) para Forms y Reports. Si queremos instalar tanto infraestructura como los contenedores en un mismo equipo tenemos que instalar cada cosa en una instancia diferente y en diferente usuario, también es necesario asignar una instancia diferente del Enterprise Manager para cada instalación, ej.: ias1, ias2.

Pueden encontrarse varios problemas al instalar tanto la infraestructura como los contenedores:

- Suele mostrarse un mensaje de que no se puede iniciar el gestor OPMN (luego se explicará en detalle para que sirve), esto se presenta cuando se están copiando los archivos en el disco duro, se debe poner continuar.
- Cuando el instalador se detiene en las configuraciones, es preferible no detener la instalación, porque se debería reiniciar todo el proceso nuevamente, desinstalando lo último y volviendo a reinstalar, sino más bien observar los archivos de logs o los mensajes que se muestran en el mismo instalador, corregir el problema y reintentar la configuración, el instalador me da la posibilidad. Los problemas frecuentes suelen ser por incompatibilidad en las librerías o el haberse olvidado instalar algún paquete.
- También en el configurador se suele detener en el inicio de OPMN, que es el Oracle ® Process Manager and Notification Server, que sirve para iniciar todos los servidores del Application Server. Para corregir este problema hay que parar el OPMN y reintentar la configuración. El OPMN se ejecuta en:

- \$ORACLE_HOME/opmn/bin/opmnctl <startall|stopall>

➤ **Después de Instalar**

Después de instalar la infraestructura se deben configurar las variables de entorno de la misma manera que en Oracle ® 10G Database Server, en cambio en el usuario que se instalan los contenedores es suficiente con configurar la variable de entorno ORACLE ®_HOME.

➤ **Iniciar y Parar el Application Server**

Primero se requiere iniciar la infraestructura, obviamente el primer paso a seguir es subir la DB, es de la misma forma como ya se detalló anteriormente. Luego es de subir el Oracle ® Internet Directory

El monitor:

- oidmon connect=<nombre de instancia de base de datos><stop|start>

La instancia del Internet Directory:

- oidctl connect=cc server=ss instance=nn <start|stop>
- donde cc=nombre de la instancia de base de datos, ss=puede ser oidldapd/oidrepld/odisrv pero en nuestro caso necesitamos iniciar el ldap y la opción sería oidldapd, nn=número de la instancia que debe ser único y es un entero. Aunque al subir la base de datos y el monitor automáticamente se sube el Internet Directory, pero también se deben subir el resto de servicios instalados con el OPMN en:

- \$ORACLE_HOME/opmn/bin/opmnctl <startall|stopall>

Por último subir el Enterprise Manager:

- emctl <start|stop> iasconsole

Luego se debe levantar los contenedores, desde el usuario que se instaló, subiendo el OPMN y el Enterprise Manager, de la misma forma descrita anteriormente.

➤ **Notas de Último Momento**

Todos los sistemas anteriormente descritos también fueron probados en Red Hat Enterprise Server 5.0 y 5.1, lográndose instalar pero con ciertas modificaciones.

Lo primero es modificar el archivo `/etc/redhat-release`, cambiar el número 5 por 4, ya que los instaladores soportan hasta Red Hat Enterprise Server 4.

Aparte de las librerías necesarias, hay que instalar las librerías `xorg-x11-deprecated-libs-6.8.2-1.EL.19.i386.rpm` (descargar de Internet) y forzar la librería `openmotif21-2.1.30-9.RHEL3.6.i386.rpm` de Red Hat Enterprise Server 3. Con estas modificaciones se puede instalar sin complicaciones.

También se probó sobre Red Hat Enterprise Server 3 y 4, resultando error en la configuración del Internet Directory del Application Server.

En Windows 2003 Server también no configuré el Internet Directory del Application Server, pero con la versión 10g 9.0.4 no resultó en ningún inconveniente. Se necesita estrictamente las condiciones para poder instalar, caso contrario el instalador no sigue.

La opción de Discoverer del Application Server no instaló sobre ninguna versión de Linux. Hay conflictos con librerías de compatibilidad de Linux.

Las versiones utilizadas son: Database Server 10g 10.2.0.1, Application Server 10g 10.1.2.0.2, Developer Suite 10g 10.1.2.0.2.

E. PROTOTIPO DE INTERFAZ DE USUARIO

Toda obra de diseño conlleva la elección de los elementos básicos que la van a formar. Una composición gráfica está destinada a representar un medio de comunicación entre personas y ordenadores, este es el caso de una interfaz de usuario.

Es decir el diseño gráfico aplicado a la construcción de interfaces Web, para conseguir un medio de interacción entre los usuarios y el conjunto de páginas de un sitio Web y las aplicaciones que corren por debajo de ellas.

➤ **Propósito**

Dar a conocer a los interesados la plantilla que regirá las aplicaciones que se desean implementar en la UTN, así como también los archivos de configuración, el mismo que servirá de base para las aplicaciones futuras.

➤ **Descripción**

Este documento presenta al interesado los siguientes aspectos:

- archivos y configuraciones necesarias para la personalización de interfaces gráficas;
- diseño de la plantilla estándar;
- funciones y procedimientos para la ejecución de los procesos básicos de la plantilla estándar.

Utilizando la plataforma Oracle ® 10g como servidor de base de datos, OAS (Oracle® Application Server 10.1.2) como servidor de aplicaciones y como IDE de programación Oracle® Developer Suite 10.1.2 con lenguaje de programación PL/SQL.

E.1. ARCHIVOS DE CONFIGURACIÓN

➤ **Visualización de Iconos**

Para visualizarlos en tiempo de ejecución haremos lo siguiente:

- Editamos el archivo `orion-web.xml` localizado en `ORA-HOME/j2ee/DevSuite/Application-deployments/forms/formsweb` y añadimos el directorio virtual donde se va encontrar los iconos:
 - `<virtual-directory virtual-path="/icons" real-path="C:MyAplicacion/iconos" />`

- Le indicamos ahora al servicio que extensión van a tener y en que directorio virtual se encuentran. Editamos el archivo Registry.dat que está en la ruta ORA-HOME/forms/java/oracle/forms/registry y añadimos o modificamos las siguientes líneas:
 - default.icons.iconpath=icons/
 - default.icons.iconextension=jpg

Si estamos trabajando con Developer Forms en tiempo de diseño, podemos observar que los botones icónicos aparecen en blanco aunque hayamos introducido la ruta correcta de donde se encuentran. La forma de implantarlos es la siguiente:

- Los nombres de los archivos icónicos no deben tener el path ni la extensión, únicamente el nombre.
- Editamos el registro de Windows y en **HKEY_LOCAL_MACHINE/Software/Oracle/HOME0** creamos la variable **UI_ICON_EXTENSION** con valor jpg ya que estamos utilizando los iconos con esta extensión. Lógicamente debemos indicar el path de los iconos en la clave **UI_ICON** (esta clave normalmente ya está creada, si no es así debemos crearla).

Con esto tendríamos configurada la visualización de íconos.

E.2. PERSONALIZACIÓN DE LA PÁGINA PRINCIPAL DE LA APLICACIÓN

➤ **Configuración del Archivo formsweb.cfg**

El archivo formsweb.cfg se encuentra ubicado en el siguiente directorio:

- toolsOracle\oracle\produc\10.2.0\db_2\forms90\server\

En este archivo se definen los valores de parámetro usados por el FormsServlet (f90servlet). Cualquiera de ellos se puede eliminar o modificar en las secciones de configuración nombradas.

A continuación se presenta un listado de los parámetros más importantes para la personalización de la página principal.

- pageTitle: Nombre del título de la página. Ejemplo:
 - # HTML page title
 - pageTitle=Aplicaciones UTN

- width: Especifica el ancho del applet del formulario, en pixeles. Por defecto es 650. Ejemplo:
 - # Forms applet parameter
 - width=980
- height: Especifica el alto del applet del formulario, en pixeles. Por defecto es 500. Ejemplo:
 - # Forms applet parameter
 - height=590
- separateFrame: Se determina si el applet aparece dentro de una ventana separada. Valores legales: Verdad o falso. Ejemplo:
 - # Forms applet parameter
 - separateFrame=false
- splashScreen: Especifica el archivo .GIF que debe aparecer antes de que aparezca el applet. Fijar a NO para no aparecer. Dejar vacío para utilizar la imagen por defecto. Para fijar el parámetro incluir el nombre del archivo (por ejemplo, myfile.gif) o la trayectoria virtual y nombre del archivo (por ejemplo, imágenes/myfile.gif). Ejemplo:
 - # Forms applet parameter
 - splashScreen=utn2.gif
- background: Especifica el archivo .GIF que debe aparecer en el fondo. Fijar a NO para ningún fondo. Dejar vacío para utilizar el fondo por defecto.
 - # Forms applet parameter
 - background=utn1.gif
- lookAndFeel: Para modificar la apariencia de la aplicación, los valores que puede tomar son:
 - generic: Apariencia típica de Windows
 - oracle: Apariencia por defecto definida por Oracle.

Ejemplo:

- # Forms applet parameter
- lookAndFeel=oracle

- colorScheme: Es el valor del parámetro lookAndFeel es oracle en colorScheme se puede definir el siguiente conjunto de colores:

- teal
- red
- titanium
- blue
- khaki
- olive
- purple

Ejemplo:

- # Forms applet parameter
- colorScheme=blue
- Logo: Especifica el archivo .GIF que debe aparecer en la barra de menú de las formas. Fijar a NO para ninguna insignia. Dejar vacío para utilizar la insignia de Oracle por defecto. Ejemplo:
 - # Forms applet parameter
 - logo=utn.gif

E.3. FUNCIONES Y PROCEDIMIENTOS PARA EJECUTAR LA PLANTILLA

➤ FUNCTION FUN_ALERTA_2BOTONES

Esta función permite establecer una alerta con 2 botones personalizada, y retorna 1, 2 o 0.

Descripción de Variables:

- pvarc2NombreAlerta Este parámetro recibe el nombre para la alerta, cuyos valores pueden ser:
 - ALE_ATENCION
 - ALE_INFORMACION
 - ALE_ERROR
- pvarc2MensajeAlerta: Este parámetro recibe el mensaje para la alerta;
- pvarc2Boton1Alerta: Este parámetro recibe el nombre del boton1;

- pvarc2Boton2Alerta: Este parámetro recibe el nombre del boton2;
- pvarc2TituloAlerta: Este parámetro recibe el título de la alerta
- InumbBanderaBoton: Esta variable obtiene el valor que retorna la alerta.

FUNCTION FUN_ALERTA_2BOTONES(

```
pvarc2NombreAlerta VARCHAR2,  
pvarc2TituloAlerta VARCHAR2,  
pvarc2MensajeAlerta VARCHAR2,  
pvarc2Boton1Alerta VARCHAR2,  
pvarc2Boton2Alerta VARCHAR2)  
RETURN NUMBER IS InumbBanderaBoton NUMBER;  
BEGIN  
    SET_ALERT_PROPERTY (pvarc2NombreAlerta, ALERT_MESSAGE_TEXT,  
        pvarc2MensajeAlerta);  
    SET_ALERT_PROPERTY (pvarc2NombreAlerta, TITLE, pvarc2TituloAlerta);  
    SET_ALERT_BUTTON_PROPERTY (pvarc2NombreAlerta, ALERT_BUTTON1,  
        LABEL,  
        pvarc2Boton1Alerta);  
    SET_ALERT_BUTTON_PROPERTY (pvarc2NombreAlerta, ALERT_BUTTON2,  
        LABEL,  
        pvarc2Boton2Alerta);  
    InumbBanderaBoton := SHOW_ALERT (pvarc2NombreAlerta);  
    IF InumbBanderaBoton = ALERT_BUTTON1 THEN RETURN 1;  
    ELSIF InumbBanderaBoton = ALERT_BUTTON2 THEN RETURN 2;  
    ELSE RETURN 0;  
    END IF;  
END;
```

➤ **FUNCTION FUN_OBTENER_FECHA_LARGA**

Esta función permite obtener la fecha actual en el siguiente formato (01 DE ENERO DEL 2007) recibiendo como parámetro la fecha actual del sistema.

Descripción de Variables:

- lvarc2FechaLarga: Variable en la que se va concatenando la fecha larga;
- lvarc2Mes: Variable que almacena el número de mes;
- lvarc2Año: Variable que almacena el año.

FUNCTION FUN_OBTENER_FECHA_LARGA (pdatFechaCorta DATE)

```
RETURN VARCHAR2 IS  
lvarc2FechaLarga VARCHAR2 (100);  
lvarc2Mes VARCHAR2 (2);  
lvarc2Año VARCHAR2(4);  
BEGIN  
    lvarc2FechaLarga := TO_CHAR (pdatFechaCorta, 'Dy') || ',';  
    lvarc2FechaLarga := lvarc2FechaLarga || TO_CHAR (pdatFechaCorta, 'DD') || ' DE '  
    lvarc2Mes := TO_CHAR (pdatFechaCorta, 'MM');  
    IF lvarc2Mes = '01' THEN
```

```
lvarc2FechaLarga := lvarc2FechaLarga || 'ENERO ';
ELSIF lvarc2Mes = '02' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'FEBRERO ';
ELSIF lvarc2Mes = '03' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'MARZO ';
ELSIF lvarc2Mes = '04' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'ABRIL ';
ELSIF lvarc2Mes = '05' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'MAYO ';
ELSIF lvarc2Mes = '06' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'JUNIO ';
ELSIF lvarc2Mes = '07' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'JULIO ';
ELSIF lvarc2Mes = '08' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'AGOSTO ';
ELSIF lvarc2Mes = '09' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'SEPTIEMBRE ';
ELSIF lvarc2Mes = '10' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'OCTUBRE ';
ELSIF lvarc2Mes = '11' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'NOVIEMBRE ';
ELSIF lvarc2Mes = '12' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'DICIEMBRE ';
END IF;
lvarc2Año := TO_CHAR (pdatFechaCorta, 'YYYY');
IF substr (lvarc2Año, 1, 1) = '2' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'DEL ' || lvarc2Año;
ELSE
    lvarc2FechaLarga := lvarc2FechaLarga || 'DE ' || lvarc2Año;
END IF; RETURN lvarc2FechaLarga;
END;

➤ FUNCTION FUN_OBTENER_HORA_ACTUAL
```

Esta función permite obtener la hora actual en el siguiente formato 21:30

Descripción de Variables:

- lvarc2Hora: Variable que almacenan las horas, en este caso en formato de 24 horas;
- lvarc2Minuto: Variable que almacenan los minutos;
- lvarc2HoraActual: Variable que almacena la hora tal como se va a mostrar.

```
FUNCTION FUN_OBTENER_HORA_ACTUAL RETURN VARCHAR2 IS
lvarc2Hora VARCHAR2 (4);
lvarc2Minuto VARCHAR2 (2);
lvarc2HoraActual VARCHAR2 (10);
BEGIN
    lvarc2Hora := TO_CHAR (SYSDATE, 'HH24');
    lvarc2HoraActual := lvarc2HoraActual || ' ' || lvarc2Hora;
    lvarc2Minuto := TO_CHAR (SYSDATE, 'MI');
    lvarc2HoraActual := lvarc2HoraActual || ':' || lvarc2Minuto;
    RETURN lvarc2HoraActual; END;
```

➤ PROCEDURE PRO_ACCIONES_TOOLBAR

Este procedimiento permite determinar que botón ha sido seleccionado de la barra y le da asigna una acción.

Descripción de Variables:

- Ivarc2NombreElemento: Esta variable sirve para recuperar el nombre del elemento seleccionado en la barra;
- Ivarc2NombreBloqueElemento: Esta variable sirve para recuperar el nombre del bloque y el elemento seleccionado en la barra;
- InumbBanderaAlerta: Esta variable sirve para obtener el valor retornado de la alerta.

PROCEDURE PRO_ACCIONES_TOOLBAR IS

Ivarc2NombreElemento VARCHAR2(30);

Ivarc2NombreBloqueElemento VARCHAR2(60);

InumbBanderaAlerta NUMBER;

BEGIN

Ivarc2NombreBloqueElemento := NAME_IN('SYSTEM.TRIGGER_ITEM');

Ivarc2NombreElemento := SUBSTR(Ivarc2NombreBloqueElemento,

INSTR(Ivarc2NombreBloqueElemento, '.') + 1);

IF(Ivarc2NombreElemento = 'CMD_GUARDAR') THEN

InumbBanderaAlerta := FUN_ALERTA_2BOTONES('ALE_INFORMACION', 'Atención UTN', 'Desea Guardar Los Cambios', 'Sí', 'No');

IF(InumbBanderaAlerta = 1) THEN

DO_KEY('COMMIT_FORM');

END IF;

ELSIF(Ivarc2NombreElemento = 'CMD_IMPRIMIR') THEN

DO_KEY('PRINT');

ELSIF (Ivarc2NombreElemento = 'CMD_LIMPIAR_FORMA') THEN

DO_KEY ('CLEAR_FORM');

:BLOQ_TOOLBAR.TXT_MOSTRAR_FECHA :=

FUN_OBTENER_FECHA_LARGA(SYSDATE);

:BLOQ_TOOLBAR.TXT_MOSTRAR_HORA :=

FUN_OBTENER_HORA_ACTUAL();

:BLOQ_TOOLBAR.TXT_MOSTRAR_USUARIO:=

get_application_property(USERNAME);

ELSIF (Ivarc2NombreElemento = 'CMD_BUSCAR') THEN

IF (name_in('SYSTEM.MODE') != 'ENTER-QUERY') THEN

DO_KEY('ENTER_QUERY');

ELSE

DO_KEY('EXECUTE_QUERY');

END IF;

ELSIF (Ivarc2NombreElemento = 'CMD_INSERTAR_REGISTRO') THEN

CREATE_RECORD;

ELSIF (Ivarc2NombreElemento = 'CMD_BORRAR_REGISTRO') THEN

InumbBanderaAlerta :=

FUN_ALERTA_2BOTONES('ALE_ATENCION', 'Atención UTN', 'Desea Eliminar El Cliente', 'Aceptar', 'Cancelar');

IF(InumbBanderaAlerta = 1) THEN

```
        DELETE_RECORD;
    END IF;
    ELSIF (lvarc2NombreElemento = 'CMD_LIMPIAR_REGISTRO') THEN
        CLEAR_RECORD;
    ELSIF (lvarc2NombreElemento = 'CMD_PRIMER_REGISTRO') THEN
        FIRST_RECORD;
    ELSIF (lvarc2NombreElemento = 'CMD_SIGUIENTE_REGISTRO') THEN
        NEXT_RECORD;
    ELSIF (lvarc2NombreElemento = 'CMD_ANTERIOR_REGISTRO') THEN
        PREVIOUS_RECORD;
    ELSIF (lvarc2NombreElemento = 'CMD_ULTIMO_REGISTRO') THEN
        LAST_RECORD;
    ELSIF (lvarc2NombreElemento = 'CMD_LISTAR') THEN
        DO_KEY('LIST_VALUES');
    ELSIF (lvarc2NombreElemento = 'CMD_EDITAR') THEN
        DO_KEY('EDIT_FIELD');
    ELSIF (lvarc2NombreElemento = 'CMD_AYUDA') THEN
        show_keys;
    ELSIF (lvarc2NombreElemento = 'CMD_SALIR') THEN
        InumbBanderaAlerta :=
        FUN_ALERTA_2BOTONES('ALE_ATENCION','Atención
        UTN','Desea Salir De La Aplicación','Sí','No');
        IF(InumbBanderaAlerta = 1) THEN
            DO_KEY ('exit_form');
        END IF;
    END IF; END;
```

➤ **PROCEDURE PRO_INFORMACION_OBJETO**

Este procedimiento permite obtener información de un objeto al pasar el mouse sobre él.
Descripción de Variables:

- varc2Objeto: Parámetro que almacena nombre de un objeto;
- varc2Informacion: Parámetro que almacena la información que va a aparecer al pasar el mouse.

PROCEDURE PRO_INFORMACION_OBJETO

(pvarc2Objeto VARCHAR2, pvarc2Informacion VARCHAR2) IS

BEGIN

```
    SET_ITEM_PROPERTY(pvarc2Objeto, TOOLTIP_TEXT, pvarc2Informacion);
    SET_ITEM_PROPERTY(pvarc2Objeto, TOOLTIP_FONT_SIZE, 800);
    SET_ITEM_PROPERTY(pvarc2Objeto, TOOLTIP_FOREGROUND_COLOR,
    'r0g50b0')
    SET_ITEM_PROPERTY(pvarc2Objeto, BACKGROUND_COLOR,
    'r180g220b180');
```

END;

➤ **PROCEDURE PRO_INFORMACION_TOOLBAR.**

Este procedimiento permite obtener información de cada uno de los objeto de la barra de herramientas al pasar el mouse.

PROCEDURE PRO_INFORMACION_TOOLBARIS

BEGIN

```
:BLOQ_TOOLBAR.TXT_MOSTRAR_FECHA:=  
FUN_OBTENER_FECHA_LARGA(SYSDATE);  
:BLOQ_TOOLBAR.TXT_MOSTRAR_HORA:=FUN_OBTENER_HORA_ACTUAL();  
:BLOQ_TOOLBAR.TXT_MOSTRAR_USUARIO:=  
get_application_property(USERNAME);  
PRO_INFORMACION_OBJETO ('BLOQ_TOOLBAR.CMD_SALIR', 'Salir');  
PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_AYUDA','Ayuda');  
PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_BUSCAR','Buscar');  
PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_IMPRIMIR','Imprimir');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_LIMPIAR_FORMA','Limpiar Forma');  
PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_LISTAR','Lista');  
PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_EDITAR','Editar');  
PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_GUARDAR','Guardar');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_INSERTAR_REGISTRO','Insertar Registro');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_LIMPIAR_REGISTRO', 'Limpiar Registro');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_BORRAR_REGISTRO', 'Borrar Registro');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_SIGUIENTE_REGISTRO','Registro Siguiente');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_ANTERIOR_REGISTRO', 'Registro Anterior');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_PRIMER_REGISTRO',' PrimerRegistro');  
PRO_INFORMACION_OBJETO  
( 'BLOQ_TOOLBAR.CMD_ULTIMO_REGISTRO','Ultimo Registro');
```

END;

➤ **PROCEDURE PRO_ALERTA**

Este procedimiento permite establecer una alerta personalizada.

Descripción de Variables:

- pvarc2NombreAlerta: Este parámetro recibe el nombre para la alerta, cuyos valores pueden ser:
 - ALE_ATENCION,
 - ALE_INFORMACION,
 - ALE_ERROR;
- pvarc2MensajeAlerta: Este parámetro recibe el mensaje para la alerta;
- pvarc2TituloAlerta: Este parámetro recibe el título de la alerta;
- InumbBanderaBoton: Esta variable obtiene el valor que retorna la alerta.

```
PROCEDURE PRO_ALERTA(pvarc2NombreAlerta VARCHAR2, pvarc2TituloAlerta  
VARCHAR2, pvarc2MensajeAlerta VARCHAR2)IS InumbBanderaBoton NUMBER;  
BEGIN  
    SET_ALERT_PROPERTY (pvarc2NombreAlerta, ALERT_MESSAGE_TEXT,  
    pvarc2MensajeAlerta);  
    SET_ALERT_PROPERTY (pvarc2NombreAlerta, TITLE, pvarc2TituloAlerta);  
    InumbBanderaBoton := SHOW_ALERT (pvarc2NombreAlerta);  
END;
```

➤ **PROCEDURE PRO_TITULO_COLOR_VENTANA**

Este procedimiento permite poner título a la ventana, además se define el color y se maximiza.

Descripción de Variables:

- pvarc2Nombre: Ventana Parámetro que recibe el nombre de la Ventana;
- pvarc2Titulo: Ventana Parámetro que recibe el título de la Ventana.

PROCEDURE PRO_TITULO_COLOR_VENTANA

(pvarc2NombreVentana VARCHAR2, pvarc2TituloVentana VARCHAR2) IS

```
BEGIN  
    PRO_VENTANA_CENTRADA('WINDOW1');  
    SET_WINDOW_PROPERTY('WINDOW1', WINDOW_STATE, MAXIMIZE);  
    SET_WINDOW_PROPERTY (pvarc2NombreVentana, TITLE,  
    varc2TituloVentana || ' Form:( ' ||  
    get_application_property(CURRENT_FORM_NAME) || ');'  
    SET_WINDOW_PROPERTY (pvarc2NombreVentana, BACKGROUND_COLOR  
    ,r200g230b210);  
END;
```

➤ **PROCEDURE PRO_VENTANA_CENTRADA**

Este procedimiento permite centrar la ventana. Descripción de Variables:

- pvarc2wiN: Parámetro que recibe el nombre de la ventana;
- lwinWinId: Variable que almacena el nombre de la ventana;
- InumbWinX: Variable para la posición en x de la ventana;
- InumbWinY: Variable para la posición en y de la ventana;
- InumbWinW: Variable para el ancho de la ventana;
- InumbWinH: Variable para el largo de la ventana;
- numbDisplayW: Variable para el ancho de la pantalla;
- InumbDisplayH: Variable para el largo de la pantalla;
- InumbHeightOffset: Variable para el largo de la ventana.

```
PROCEDURE PRO_VENTANA_CENTRADA( pvarc2Win VARCHAR2 ) IS
lwinWinId window;
lnumbWinX NUMBER;
lnumbWinY NUMBER;
lnumbWinW NUMBER;
lnumbWinH NUMBER;
lnumbDisplayW NUMBER;
lnumbDisplayH NUMBER;
lnumbHeightOffset NUMBER := 0;
BEGIN
IF Get_Application_Property(USER_INTERFACE)='MSWINDOWS' THEN
    lnumbHeightOffset := .05; -- inches;
END IF;
lwinWinId := FIND_WINDOW(pvarc2Win);
IF ID_NULL(lwinWinId) THEN RETURN;
END IF;
lnumbDisplayH := TO_NUMBER(GET_APPLICATION_PROPERTY(DISPLAY_HEIGHT));
lnumbDisplayW := TO_NUMBER(GET_APPLICATION_PROPERTY(DISPLAY_WIDTH));
lnumbWinX := GET_WINDOW_PROPERTY(lwinWinId, X_POS);
lnumbWinY := GET_WINDOW_PROPERTY(lwinWinId, Y_POS);
lnumbWinW := GET_WINDOW_PROPERTY(lwinWinId, WIDTH);
lnumbWinH := GET_WINDOW_PROPERTY(lwinWinId, HEIGHT);
lnumbWinH := lnumbWinH+100;
IF( lnumbWinW >= lnumbDisplayW ) THEN lnumbWinX := 0;
ELSE lnumbWinX := (lnumbDisplayW - lnumbWinW) / 2;
END IF;
IF (lnumbWinH >= lnumbDisplayH ) THEN lnumbWinY := 0;
ELSE lnumbWinY := (lnumbDisplayH - lnumbHeightOffset - lnumbWinH) / 2;
END IF;
SET_WINDOW_PROPERTY(lwinWinId, X_POS, lnumbWinX-20);
SET_WINDOW_PROPERTY(lwinWinId, Y_POS, lnumbWinY-55);
SHOW_WINDOW(lwinWinId);END;
```

F. MANUAL DE USUARIO

A través de este manual el usuario podrá utilizar el sistema desarrollado de forma correcta, para lo cual debe tener en cuenta las siguientes indicaciones:

- Ingresar a un navegador puede ser este (Internet Explorer, Firefox o Google Chrome);
- Ingresar al portal de la universidad <http://www.utn.edu.ec/>;
- Dar clic en el enlace del Sistema de Información Integrado UTN;
- Aparece esta dirección: <http://172.20.1.173:7778/forms/frmservlet?config=utn>
- Logearse con su cuenta y clave de Usuario.

Tome en cuenta que en todas las pantallas del sistema aparecerá la siguiente pantalla:



Figura F.1. Barra de Herramientas.

Fuente: [Propia].

- Limpiar
- Guardar
- Imprimir
- Buscar
- Listar
- Escribir
- Insertar Registro
- Borrar
- Eliminar Registro
- Primer Registro
- Anterior Registro
- Siguiente Registro
- Ultimo Registro
- Ayuda en botones de control
- Salir
- Usuario conectado
- Formulario Actual.

F.1. INGRESO AL SISTEMA INFORMÁTICO INTEGRADO UTN

En la siguiente figura Se muestra la pantalla de login, en la cual se ingresa el usuario, contraseña y la base de datos a la que se van a conectar en este caso es Servidor.

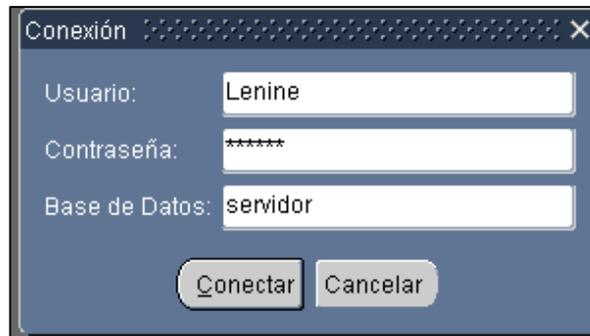


Figura F.2. Pantalla de Ingreso al Sistema Integrado.
Fuente: [Propia].

F.2. INGRESO AL SISTEMA PARA LA GESTIÓN DE LA INVESTIGACIÓN

Una vez logeado el usuario se presenta la siguiente pantalla, en esta se debe seleccionar las opciones Inicio, Gestión Científico – Tecnológico y Centro de Investigación Científica - Tecnológica.



Figura F.3. Pantalla del Sistema Integrado de la UTN.
Fuente: [Propia].

Cuando se haya realizado los pasos anteriores se puede visualizar la pantalla del Sistema de Gestión de la Investigación, la cual tiene los siguientes menús: parámetros, mantenimiento, procesos y reportes. Estos menús serán detallados a continuación:



Figura F.4. Pantalla del Sistema de Gestión de la Investigación.
Fuente: [Propia].

F.2.1. Sección Parámetro

En la sección de parámetros se despliegan todos los atributos iniciales que necesitamos para poner en funcionamiento el sistema.



Figura F.5. Pantalla Parámetros del Sistema.
Fuente: [Propia].

➤ Áreas de Investigación

En esta sección se pueden crear las áreas de investigación, por ejemplo: Ambiente y Energía, Desarrollo Agropecuario Sustentable, Salud, etc.

Para crear una nuevo área de investigación hacer clic en el botón: Insertar Registro. Ingresar el estado, la fecha de vigencia, la descripción, formato y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.



Figura F.6.Pantalla Áreas de Investigación.
Fuente: [Propia].

➤ Coberturas de Proyecto

En esta sección se ingresa la cobertura del proyecto, por ejemplo: Mundial, Continental, Nacional, Regional, etc.

Para ingresar una nueva cobertura hacer clic en el botón: Insertar Registro. Ingresar el estado, la descripción, formato y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

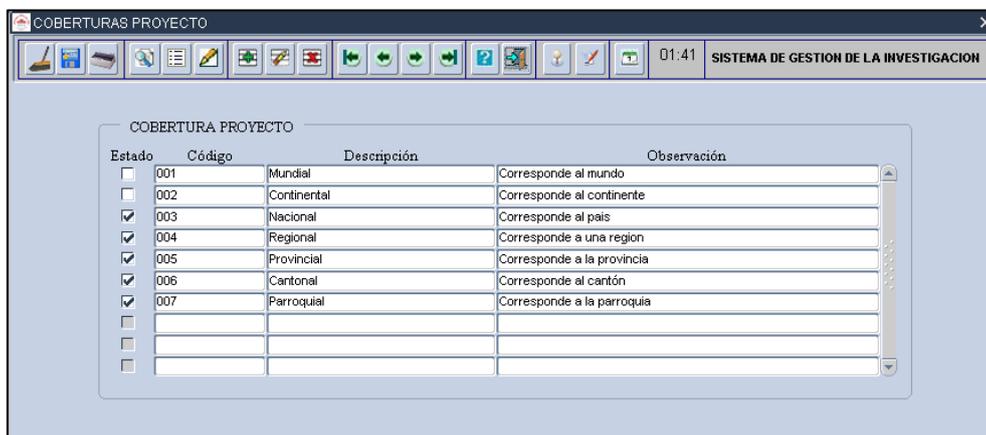


Figura F.7.Pantalla Cobertura Proyecto.
Fuente: [Propia].

➤ Definiciones Investigación

En esta sección se ingresa la cobertura del proyecto, por ejemplo: Mundial, Continental, Nacional, Regional, etc.

Para ingresar una definición de investigación hacer clic en el botón: Insertar Registro. Ingresar el estado, la descripción, formato y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

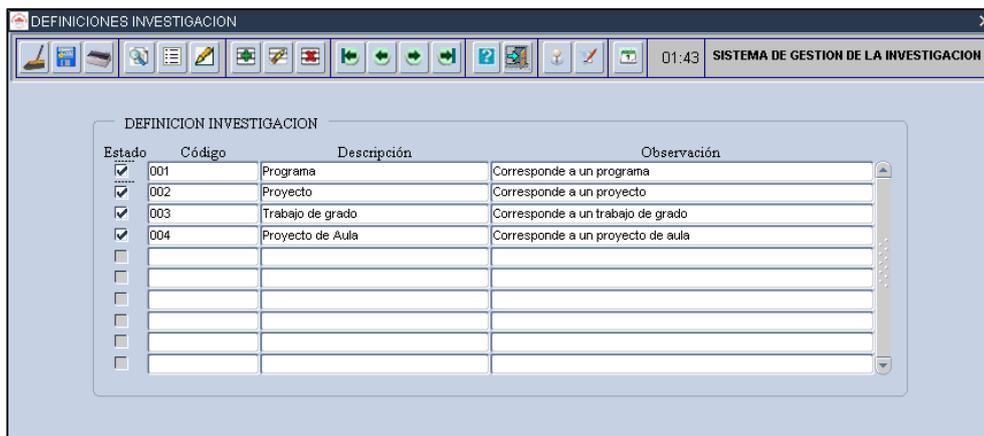


Figura F.8. Pantalla Definiciones de Investigación.
Fuente: [Propia].

➤ Líneas de Investigación

Ejes Transversales: En esta sección no se puede agregar ni borrar registros ya que estos son ingresados a través de otro formulario, se los presenta para poderlos cruzar con las áreas y líneas de investigación.

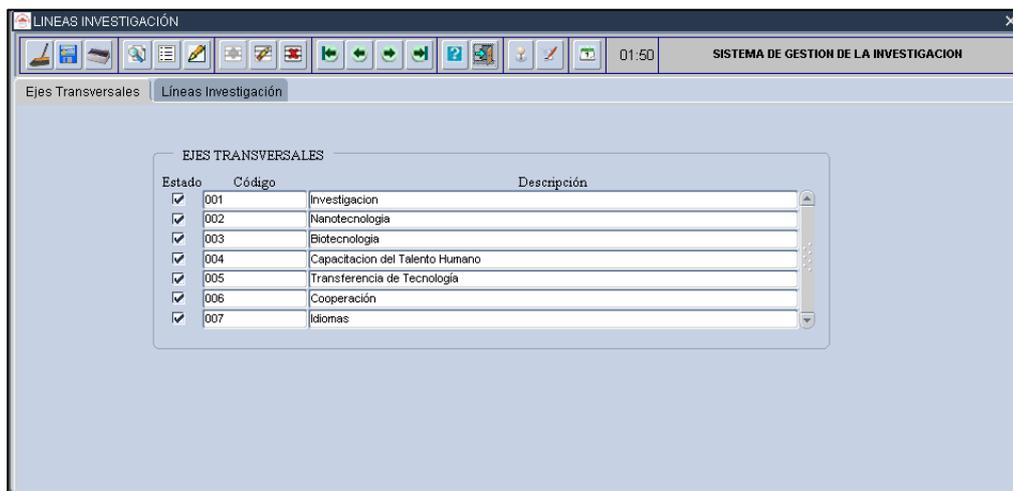


Figura F.9. Pantalla Cruce de Ejes Transversales.
Fuente: [Propia].

Líneas de Investigación: En esta sección se ingresa las líneas de investigación para lo cual:

- Dar clic en el árbol de la parte izquierda del formulario que contiene el tipo de formato para el que se va a hacer el ingreso, estos formatos son: SENESCYT, SEMPLADES, etc.
- Seleccionar el área de investigación para la línea de investigación a ingresarse.
- Dar clic en el botón insertar registro e ingresar: el estado, la descripción, formato y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

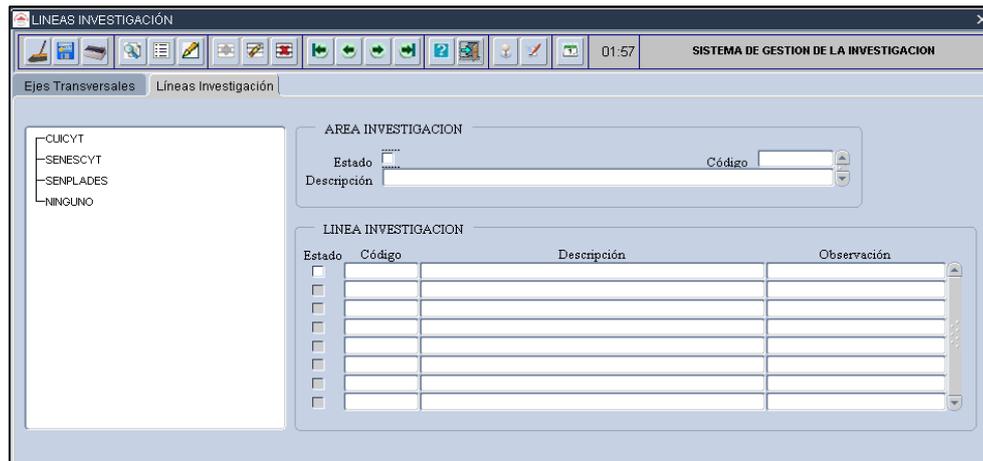


Figura F.10. Pantalla Líneas de Investigación.
Fuente: [Propia].

➤ Políticas de Investigación

En esta sección el usuario puede ingresar las políticas de investigación como por ejemplo: Convocatoria a presentación de proyectos.

Para crear una nueva política de investigación de clic en el botón insertar nuevo registro: activar la casilla de control estado para determinar a este nuevo registro activo, ingrese la descripción.

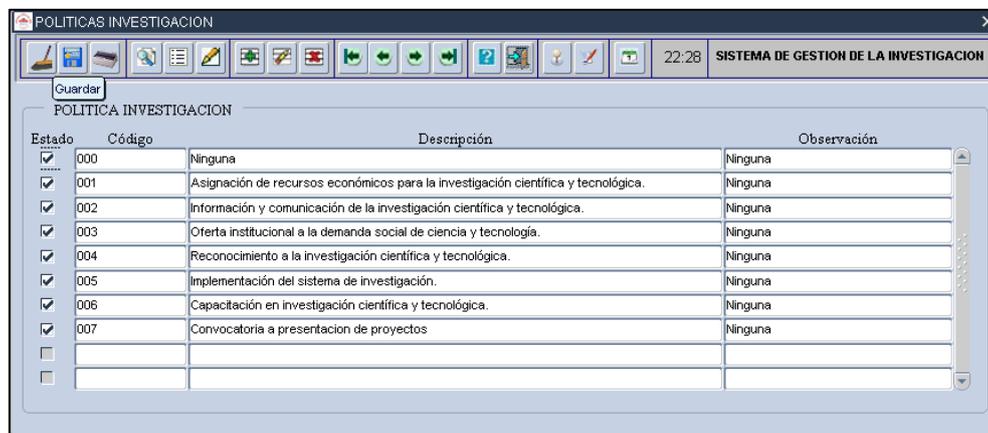


Figura F.11. Pantalla Políticas de Investigación.
Fuente: [Propia].

➤ **Categoría Proyecto**

En esta sección se pueden crear categorías de proyectos, por ejemplo: Investigador, Docente, Estudiante o Empleado.

Para crear una nueva categoría de proyecto hacer clic en el botón insertar registro. Ingresar: el estado, la descripción y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

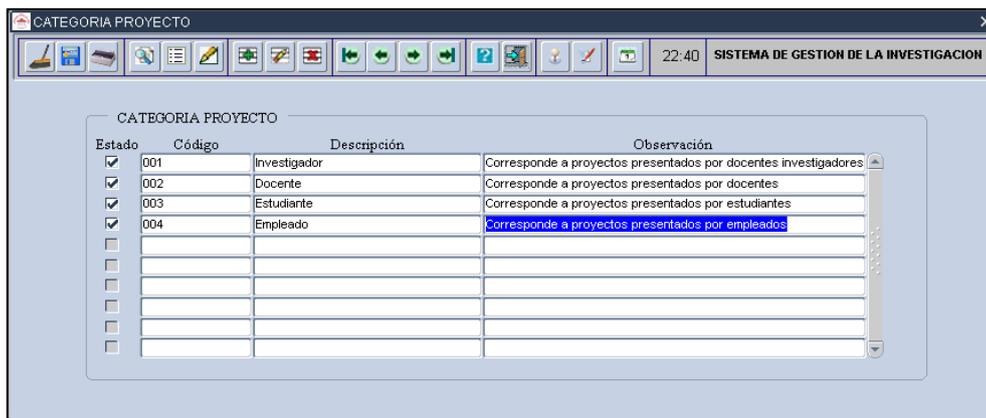


Figura F.12. Pantalla Categoría Proyecto.
Fuente: [Propia].

➤ **Ejes Transversales**

En esta sección se pueden crear ejes transversales de investigación, por ejemplo: Investigación, Nanotecnología, Biotecnología, Idiomas, etc.

Para crear un nuevo eje transversal dar clic en el botón insertar registro. Ingresar: el estado, la descripción y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

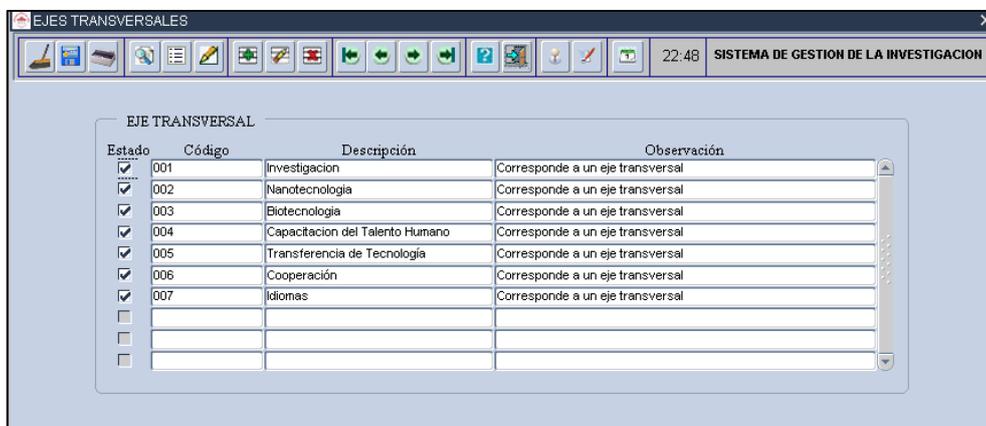


Figura F.13. Pantalla Ejes Transversales.
Fuente: [Propia].

➤ **Estados Informe**

En esta sección se pueden crear estados de informe de avance de proyectos, por ejemplo: Presentado, Revisado, Aprobado, etc.

Para crear un nuevo estado de informe dar clic en el botón insertar registro. Ingresar: el estado, la descripción y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

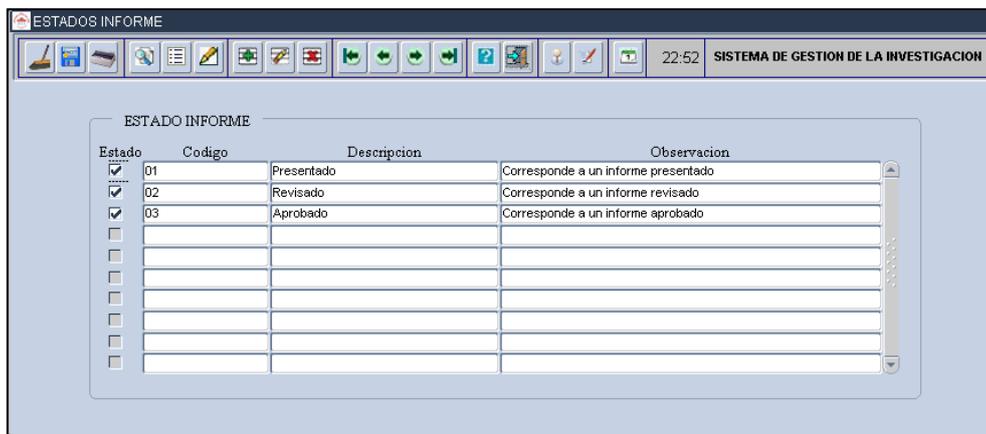


Figura F.14. Pantalla Estados Informe.
Fuente: [Propia].

➤ **Estados Proyecto**

En esta sección se pueden crear estados de proyecto, por ejemplo: Presentado, Revisado, Aprobado, Suspendido, etc.

Para crear un nuevo estado dar clic en el botón insertar registro. Ingresar: el estado, la descripción y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.

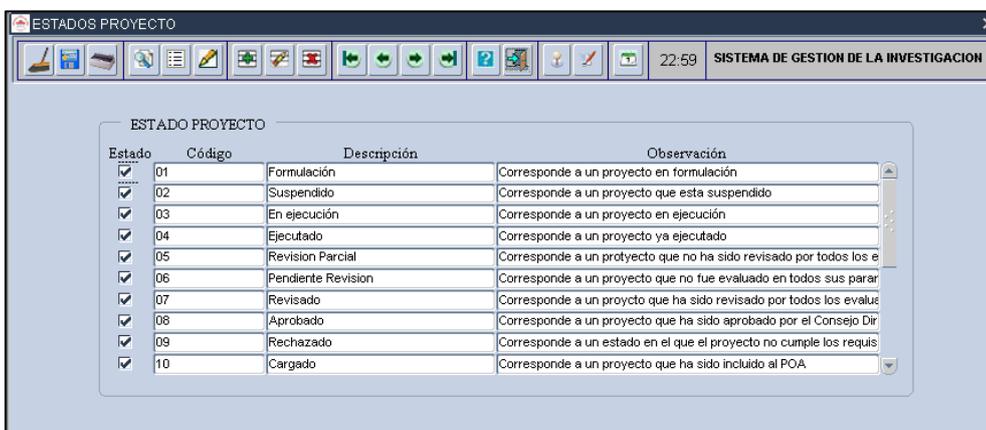


Figura F.15. Pantalla Estados Proyecto.
Fuente: [Propia].

➤ **Formato Proyecto**

Vigencia Formato: En esta sección se puede ingresar formatos de proyectos de acuerdo a un tipo de formato como por ejemplo SENESCYT. Para ingresar un nuevo formato:

- Seleccionamos el tipo de formato del cual se quiere ingresar.
- Registramos los campos de: vigente, fecha vigencia, definición y descripción.

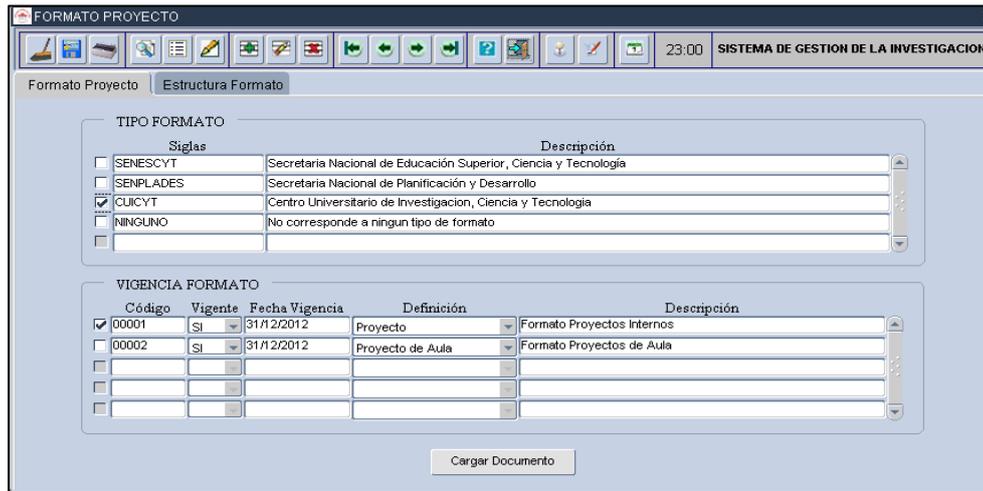


Figura F.16. Pantalla Vigencia Formato.
Fuente: [Propia].

Estructura Formato: Cuando se haya ingresado los datos del formato que se está registrando, damos clic en la casilla de control del bloque de datos vigencia formato y el sistema automáticamente muestra la pantalla de formato proyecto. Para ingresar damos clic en insertar registro e ingresamos el estado, la descripción y la observación si es que la hubiese. La parte izquierda del formulario muestra cómo se va armando la estructura del formato de proyecto.

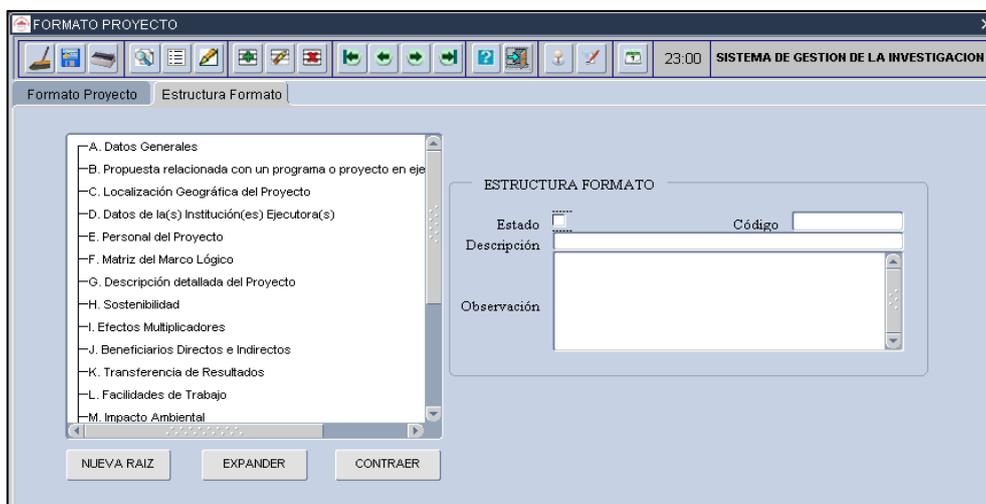


Figura F.17. Pantalla Estructura Formato.
Fuente: [Propia].

➤ Niveles de Participación

En esta sección se pueden crear niveles de participación para las instituciones asociadas al proyecto, por ejemplo: Ejecutora, Cofinanciadora, etc.

Para crear un nuevo nivel de participación dar clic en el botón insertar registro. Ingresar: el estado, la descripción y una observación en el caso que sea necesario, para guardar hacer clic en el botón guardar.



Figura F.18. Pantalla Niveles de Participación.
Fuente: [Propia].

➤ Parámetros de Evaluación

A través de este formulario se puede definir los parámetros y sub parámetros de evaluación para cada una de las definiciones de investigación (programas, proyectos, trabajos de grado o proyectos de aula) previamente definidas.

Cada vez que se quiera ingresar un parámetro o sub parámetro de evaluación tiene que seguir los siguientes pasos:

- Seleccionar del árbol de datos ubicado en la parte izquierda del formulario una definición de investigación.
- Ubicar el cursor del mouse el en el bloque de datos denominado “parámetros de evaluación”,
- Dar clic en el botón insertar registro.
- Activar la casilla de control denominada estado, ingresar el valor máximo, la descripción y la observación si es que hubiera.
- Dar clic en el botón guardar.

- Ubicar el cursor del mouse en el bloque de datos denominado “sub parámetros de evaluación”.
- Nuevamente damos clic en el botón insertar registro.
- Activamos la casilla de control estado, ingresamos la descripción y el valor máximo. El campo total se llenara automáticamente.
- Presionamos el botón guardar.

Figura F.19. Pantalla Parámetros de Evaluación.
Fuente: [Propia].

➤ Períodos Proyecto

En esta sección se puede crear periodos para la ejecución de los proyectos, por ejemplo: primero, segundo, tercero, etc.

Para crear un nuevo período: hacer clic en el botón insertar registro, ingresar la descripción, ingresar una observación en el caso que sea necesario y guardar haciendo clic en el botón guardar.

Código	Descripción	Observación
01	Primero	Corresponde al primer año o período
02	Segundo	Corresponde al segundo año o período
03	Tercero	Corresponde al tercer año o período
04	Cuarto	Corresponde al cuarto año o período
05	Quinto	Corresponde al quinto año o período
06	Sexto	Corresponde al sexto año o período
07	Septimo	Corresponde al séptimo año o período
08	Octavo	Corresponde al octavo año o período
09	Noveno	Corresponde al noveno año o período
10	Décimo	Corresponde al décimo año o período

Figura F.20. Pantalla Períodos Proyecto.
Fuente: [Propia].

➤ Porcentajes de Evaluación

En esta sección se puede ingresar el porcentaje acumulado que debe obtener un proyecto al ser evaluado con los parámetros y sub parámetros de evaluación, todo esto de acuerdo a la definición de investigación.

Para crear un nuevo porcentaje de evaluación hacer clic en el botón insertar registro, activar la casilla estado, ingresar el porcentaje , seleccionar la definición de investigación, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

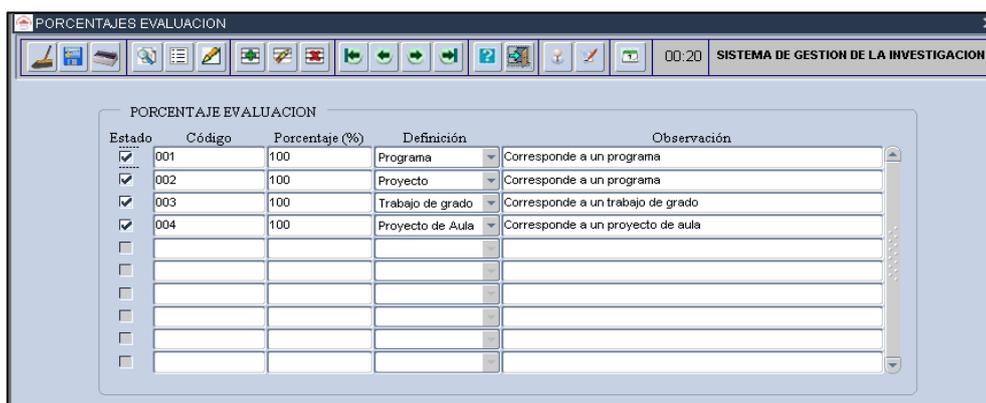


Figura F.21. Pantalla Porcentajes de Evaluación.
Fuente: [Propia].

➤ Sectores de Impacto

En esta sección se puede crear los sectores de impacto en los que se pueden desarrollar los proyectos, por ejemplo: Sector Social, Sector Productivo, etc.

Para crear un nuevo sector de impacto: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.



Figura F.22. Pantalla Sectores de Impacto.
Fuente: [Propia].

➤ **Tipo Convocatoria**

En esta sección se puede crear los tipos de convocatoria, por ejemplo: convocatoria a presentación de proyectos internos, proyectos externos, semilleros, etc.

Para crear un nuevo tipo de convocatoria: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

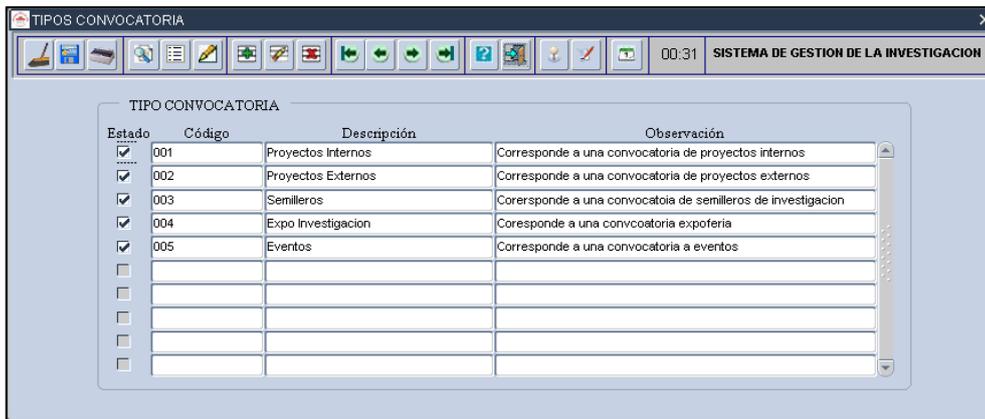


Figura F.23. Pantalla Tipo Convocatoria.
Fuente: [Propia].

➤ **Tipo Documento Respaldo**

En esta sección se puede crear tipos de documentos de respaldo, por ejemplo: oficio, hoja de vida, proyecto, formato de proyecto, etc.

Para crear un nuevo tipo de documento de respaldo: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

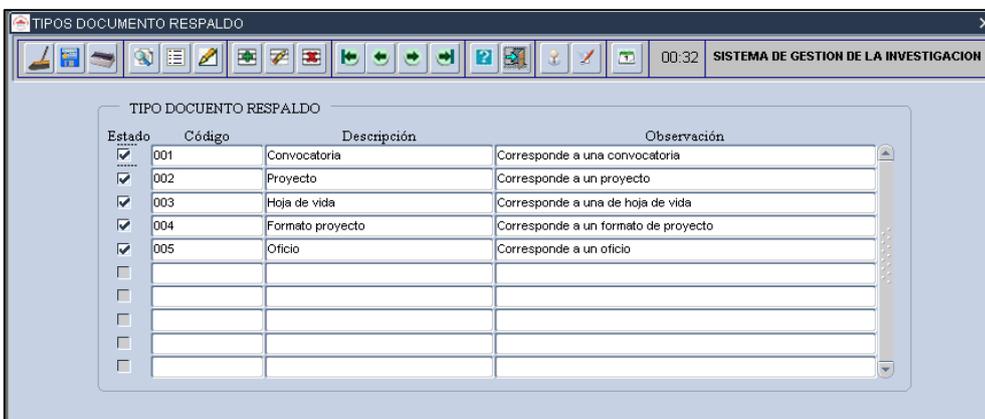


Figura F.24. Pantalla Tipo Documento Respaldo.
Fuente: [Propia].

➤ **Tipo Evento**

En esta sección se puede crear tipos de eventos, por ejemplo: congreso, foro, conferencia, seminario, capacitación video conferencia, etc.

Para crear un nuevo tipo de evento: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

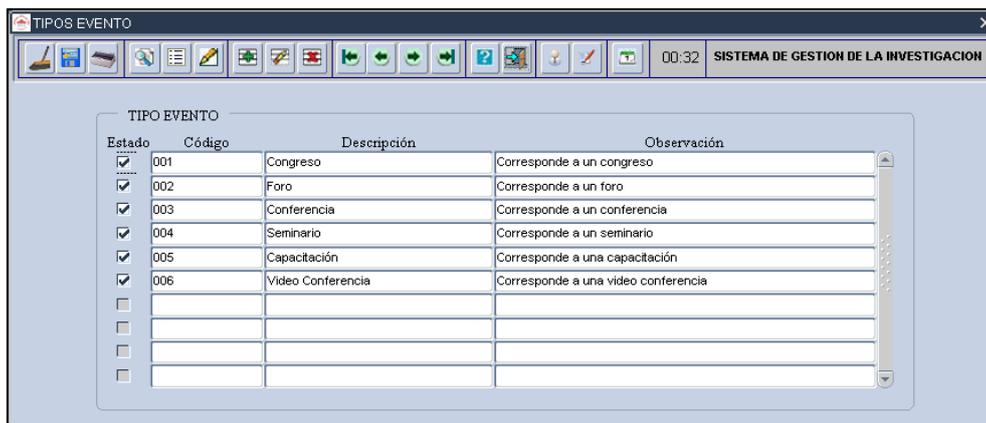


Figura F.25. Pantalla Tipo Evento.
Fuente: [Propia].

➤ **Tipo Formato Proyecto**

En esta sección se puede crear tipos de formato de proyectos, por ejemplo: SENESCYT, SEMPLADES, CUICT, etc.

Para crear un nuevo tipo de formato de proyecto: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar las siglas, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

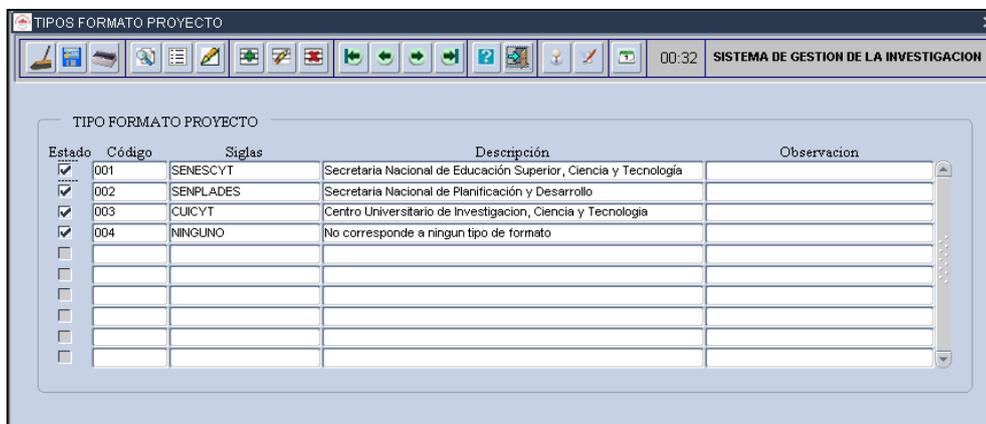


Figura F.26. Pantalla Tipo Formato Proyecto.
Fuente: [Propia].

➤ Tipo Informe Proyecto

En esta sección se puede crear tipos de informes de proyecto, por ejemplo: mensual, trimestral, anual, etc.

Para crear un nuevo tipo de informe de proyecto: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

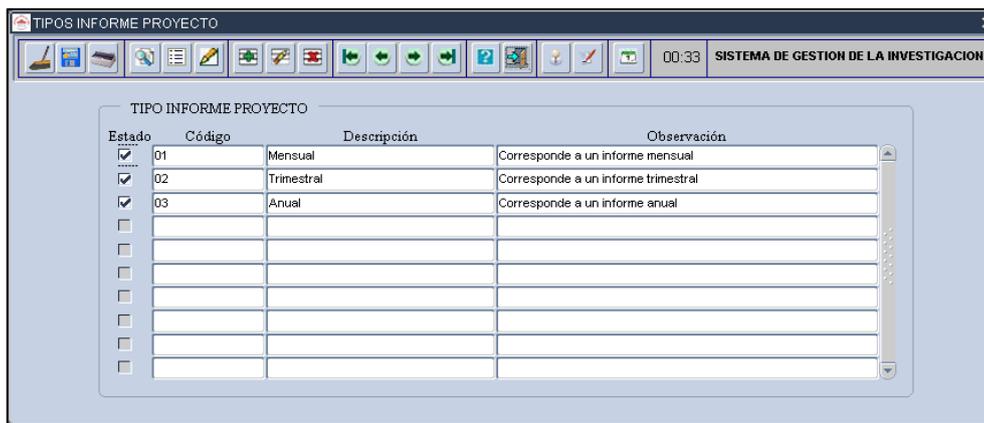


Figura F.27. Pantalla Tipo Informe Proyecto.

Fuente: [Propia].

➤ Tipo Objetivo

En esta sección se puede crear los tipos de objetivos que servirán para el ingreso del marco lógico del proyecto, estos tipos de objetivos son: fin propósito, componente, actividad, etc.

Para crear un nuevo tipo de objetivo: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.



Figura F.28. Pantalla Tipo Objetivo.

Fuente: [Propia].

➤ Tipo Participación

En esta sección se puede crear el tipo de participación que tendrá una persona (investigador, docente, estudiante, empleado) en un proyecto, por ejemplo: director del proyecto, tesista, etc.

Para crear un nuevo tipo de participación: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.



Figura F.29. Pantalla Tipo Participación.

Fuente: [Propia].

➤ Tipo Proyecto

En esta sección se puede crear los tipos de proyecto, por ejemplo: nuevo, incluido, complementario, sustitutivo, etc.

Para crear un nuevo tipo de proyecto: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.



Figura F.30. Pantalla Tipo Proyecto.

Fuente: [Propia].

➤ **Tipo Rubro**

En esta sección se puede crear los tipos de rubros de acuerdo a un formato de proyecto, ya que cada formato de proyecto maneja diferentes tipos de rubros.

Para crear un nuevo tipo de convocatoria: hacer clic en el botón insertar registro, activar la casilla de estado, ingresar la descripción, seleccionar el formato, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

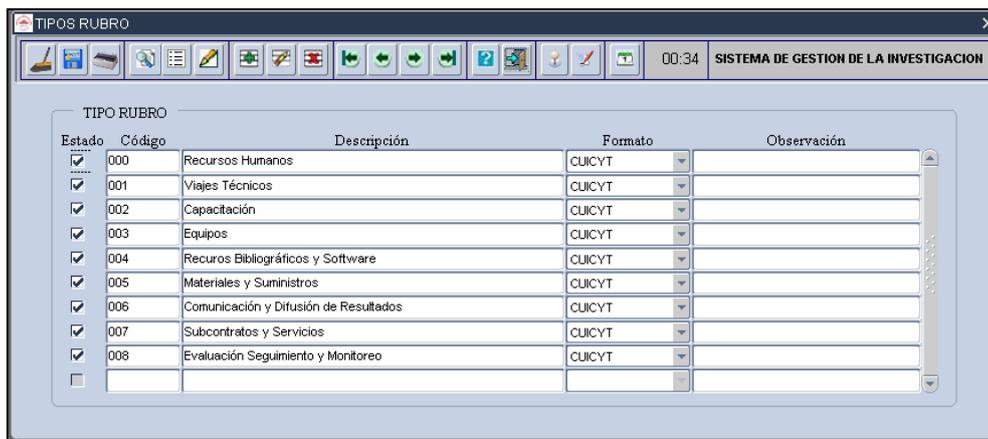


Figura F.31. Pantalla Tipo Rubro.
Fuente: [Propia].

➤ **Tipología Proyecto**

En esta sección se puede crear la tipología del proyecto, por ejemplo: investigación, desarrollo tecnológico innovación, etc.

Para crear una nueva tipología de proyecto: hacer clic en el botón insertar registro, ingresar la descripción, ingresar una observación si la hay y guardamos dando clic en el botón guardar.

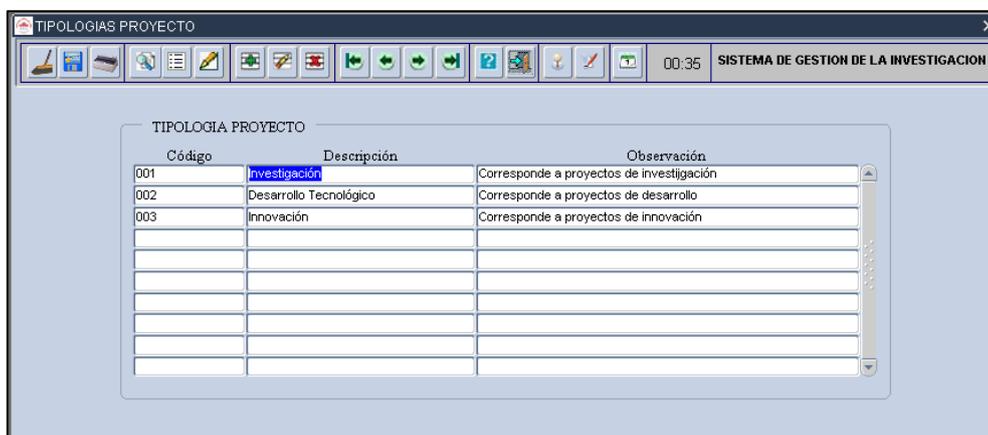


Figura F.32. Pantalla Tipología Proyecto.
Fuente: [Propia].

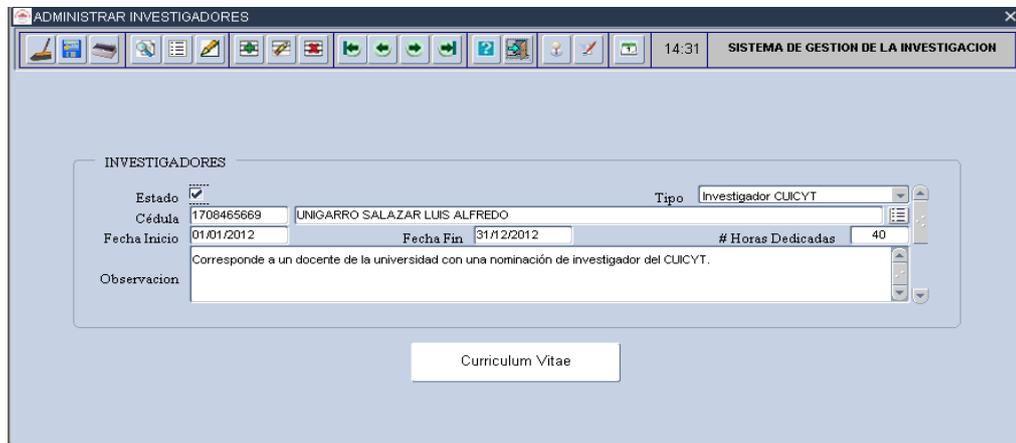
F.2.2. Sección Mantenimiento

Este menú contiene opciones que permiten administrar: investigadores, convocatorias, evaluadores, proyectos asignados a los evaluadores, el procedimiento a seguir para aplicar a una convocatoria.

➤ Investigadores

1. Ingresar Investigadores

En esta sección se puede ingresar investigadores y para ingresar un investigador: dar clic en el botón “insertar registro”, llenar los datos necesarios: estado, tipo, cedula, fecha inicio, fecha fin, etc. y guardar los datos.

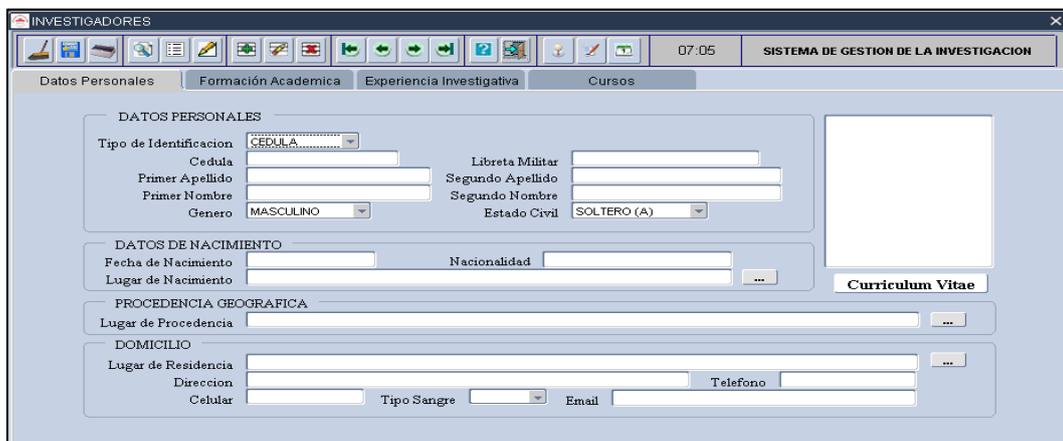


The screenshot shows a web application window titled 'ADMINISTRAR INVESTIGADORES'. The window has a toolbar with various icons and a system tray showing the time '14:31' and the text 'SISTEMA DE GESTION DE LA INVESTIGACION'. The main content area is titled 'INVESTIGADORES' and contains a form with the following fields: 'Estado' (checked), 'Tipo' (dropdown menu with 'Investigador CUICT' selected), 'Cédula' (text input with '1708465669'), 'UNIGARRO SALAZAR LUIS ALFREDO' (text input), 'Fecha Inicio' (text input with '01/01/2012'), 'Fecha Fin' (text input with '31/12/2012'), and '# Horas Dedicadas' (text input with '40'). Below these fields is an 'Observacion' text area containing the text 'Corresponde a un docente de la universidad con una nominación de investigador del CUICT.'. At the bottom center of the form is a 'Curriculum Vitae' button.

Figura F.33. Pantalla Ingresar Investigador.
Fuente: [Propia].

2. Currículo Investigador

En esta sección se puede ingresar el currículum del investigador: datos personales.



The screenshot shows a web application window titled 'INVESTIGADORES'. The window has a toolbar and a system tray showing the time '07:05' and the text 'SISTEMA DE GESTION DE LA INVESTIGACION'. The main content area is titled 'INVESTIGADORES' and has a tabbed interface with four tabs: 'Datos Personales', 'Formación Académica', 'Experiencia Investigativa', and 'Cursos'. The 'Datos Personales' tab is active and contains the following fields: 'Tipo de Identificación' (dropdown menu with 'CEDULA' selected), 'Cedula' (text input), 'Libreta Militar' (text input), 'Primer Apellido' (text input), 'Segundo Apellido' (text input), 'Primer Nombre' (text input), 'Segundo Nombre' (text input), 'Genero' (dropdown menu with 'MASCULINO' selected), and 'Estado Civil' (dropdown menu with 'SOLTERO (A)' selected). Below these fields are sections for 'DATOS DE NACIMIENTO' (Fecha de Nacimiento, Lugar de Nacimiento, Nacionalidad), 'PROCEDENCIA GEOGRAFICA' (Lugar de Procedencia), and 'DOMICILIO' (Lugar de Residencia, Direccion, Celular, Tipo Sangre, Email, Telefono). A 'Curriculum Vitae' button is located at the bottom right of the form.

Figura F.34. Pantalla Currículo Investigador.
Fuente: [Propia].

➤ Convocatorias

1. Realizar Convocatoria

En este formulario podemos crear una convocatoria (proyectos internos, proyectos externos, semilleros de investigación, eventos) de acuerdo a las políticas, áreas y líneas de investigación establecidas por el departamento.

Para crear una convocatoria:

- Hacer clic en el botón insertar registro.
- Ingresar los datos de la convocatoria: estado, período académico, desde, hasta, tipo, responsable, formato, lugar, tema y descripción. Guardamos dando clic en el botón guardar.
- Ubicamos el cursor del ratón en el bloque de datos “política de investigación” y damos clic en el botón insertar registro pero esta vez para insertar una política de investigación.
- Para seleccionar la política de investigación se tiene que mostrar la lista de valores dando clic en el botón “listar”, seleccionamos y guardamos dando clic en el botón “Guardar”.
- Cuando se haya ingresado la política de investigación procedemos a ingresar las áreas y líneas de investigación de la misma manera.

The screenshot shows a web application window titled 'CONVOCATORIAS' with a toolbar and a system clock showing 11:02. The main form is divided into several sections:

- CONVOCATORIA:** Fields for Estado (checked), Código (000000001), Período Académico (2011-2012), Desde (10/02/2012), Hasta (31/10/2012), Tipo (Proyectos Internos), Responsable/s (Lennin Escobar), Formato (CUICYT), Formato Proyectos Internos, Lugar (CUICYT), Tema (Convocatoria a presentación de proyectos internos 2012), and Descripción (El Vicerrectorado Académico a través del Centro de Investigación Científica y Tecnológica (CUICYT), convoca a todos los docentes de la UTN a presentar proyectos de investigación).
- POLITICA DE INVESTIGACION:** A list box containing 'Convocatoria a presentación de proyectos'.
- AREAS DE INVESTIGACION:** A list box containing 'Desarrollo agropecuario sustentable', 'Ambiente y Energía', 'Desarrollo de Sistemas, Mecatronica y Comunicacion', 'Desarrollo Industrial y Tecnologico', and 'Educacion, Cultura y Deportes'.
- LINEAS DE INVESTIGACION:** A list box containing 'Produccion y Producctividad'.

Figura F.35. Pantalla Convocatorias.
Fuente: [Propia].

2. Procedimiento Convocatoria

En esta sección se puede ingresar el procedimiento a seguir para aplicar en una convocatoria.

Para ingresar un registro: hacer clic en el botón insertar registro, ingresar la descripción, ingresar una observación en el caso que sea necesario y para guardar hacer clic en el botón guardar.

Código	Descripción	Observación
000000001	El CUICT previa aprobación del Consejo Directivo, convoca a los docentes de la Universidad Técnica del Norte a la present	Ninguna
000000002	El CUICT facilitará los formatos para la presentación de los proyectos.	Ninguna
000000003	La promoción de la convocatoria se realizará a través de la página web de la institución, e-mail de los docentes universitari	Ninguna
000000004	Los proyectos de investigación se enmarcarán en las áreas y líneas prioritarias de investigación de la institución y nacional	Ninguna
000000005	Los proyectos de investigación se recibirán vía internet a los siguientes correos electrónicos cuicyt@utn.edu, cuicyt@gr	Ninguna
000000006	El Consejo Directivo del CUICT nominará una comisión técnica para la revisión y selección de los proyectos.	Ninguna
000000007	El Consejo Directivo del CUICT aprobará los proyectos de investigación previo informe de la Comisión Técnica.	Ninguna
000000008	El CUICT proporcionará el apoyo administrativo y logístico para el desarrollo de los proyectos aprobados.	Ninguna
000000009	El monitoreo y seguimiento técnico y financiero del proyecto lo realizará los funcionarios del CUICT.	Ninguna
000000010	Los informes parciales, informe final y el artículo científico correspondiente se recibirán en el CUICT de acuerdo al cron	Ninguna

Figura F.36. Pantalla Procedimiento Convocatorias.

Fuente: [Propia].

3. Asignar Evaluador Convocatoria

En esta sección se puede ingresar evaluadores de proyectos, asignándolos una convocatoria de proyectos realizada. Estos evaluadores pueden ser un investigador, un docente u otra persona externa al universidad cuyo perfil este enmarcado a una de las áreas de investigación de la convocatoria realizada.

Para ingresar un evaluador hay que:

- Definir el tipo de evaluador seleccionando una de las opciones presentadas en “tipo evaluador”.
- Damos clic en el botón ingresar registro.
- Ingresamos los datos necesarios como: cedula, nombre, estado, área de investigación y observación en caso de que exista.
- Guardamos dando clic en el botón guardar.

The screenshot shows a software window titled 'EVALUADORES' with a standard Windows-style title bar. The window contains a form with three main sections. The first section, 'CONVOCATORIA', contains several input fields: 'Estado' with a checked checkbox, 'Período Académico' with the value '2011-2012', 'Fecha Convocatoria' with '10/02/2012', 'Tema' with 'Convocatoria a presentación de proyectos internos 2012', 'Tipo Convocatoria' with a dropdown menu showing 'Proyectos Internos', and 'Formato Proyecto' with 'CUICT'. The second section, 'TIPO EVALUADOR', has three radio buttons: 'Investigador', 'Docente' (which is selected), and 'Otro'. The third section, 'EVALUADOR', includes 'Cédula' (1001506052), 'Nombre' (JUAN CARLOS GARCIA PINCHAO), 'Estado' (checked), 'Área Investigación' (Desarrollo de Sistemas, Mecatronica y Comunicacion), and an 'Observación' field containing the text 'ninguna'. The window also features a toolbar with various icons and a status bar at the bottom right showing the time '14:02' and the system name 'SISTEMA DE GESTION DE LA INVESTIGACION'.

Figura F.37. Pantalla Asignar Evaluador Convocatoria.
Fuente: [Propia].

4. Asignar Proyecto Evaluador

En esta sección se puede asignar proyectos de investigación a los evaluadores de acuerdo a su área de investigación.

Para asignar un proyecto:

- Seleccionar la convocatoria.
- Ubicamos el cursor del mouse en el bloque de datos “evaluadores” y damos clic en el botón “insertar registro”.
- Dar clic en el boton “listar” para desplegar la lista de valores de evaluadores.
- Seleccionar un evaluador de la lista y guardamos dando clic en el botón “guardar”
- Ahora ubicamos el cursor del mouse en el bloque de datos “proyectos asignados” y damos clic en el botón “insertar registro”.
- Desplegamos la lista de valores dando clic en el botón “listar”, la lista mostrara datos siempre y cuando exista un proyecto en el área de investigación del evaluador.
- Seleccionamos un proyecto de la lista y procedemos a guardar dando clic en el botón “guardar”.

CONVOCATORIAS			
Estado	<input checked="" type="checkbox"/>	Tipo Convocatoria	Proyectos Internos
Periodo Academico	2011-2012	Fecha Convocatoria	10/02/2012
		Formato Proyecto	CUICT
Tema	Convocatoria a presentacion de proyectos internos 2012		

EVALUADORES		
Cedula	Nombres	Area Investigacion
1001506052	JUAN CARLOS GARCIA PINCHAO	Desarrollo de Sistemas, Mecatronica y Comunicacion

PROYECTOS ASIGNADOS		
Estado	Titulo	Observación
Asignado	GESTION DE LA INVESTIGACION	NINGUNA

Figura F.38. Pantalla Asignar Proyecto Evaluador.
Fuente: [Propia].

➤ Semilleros

1. Ingresar Semillero

En esta sección se puede ingresar propuestas de proyectos de aula, los mismos que serán partícipes en los encuentros de semilleros de investigación.

Para ingresar un proyecto de aula:

- En cada una de las pestañas del formulario es necesario ir guardando los cambios realizados.
- Seleccionar la convocatoria.
- Ingresar los datos solicitados.
- Dar clic en el botón “Área y Línea Investigación” para definir las mismas.
- Ingresar los integrantes del proyecto de aula.
- Ingresar la descripción del proyecto de aula.
- Ingresar los costos que demandan.
- Guardar los datos registrados en el formulario dando clic en el botón “Guardar”.

Figura F.39. Pantalla Ingresar Semillero.
Fuente: [Propia].

2. Evaluar Semillero

En esta sección se pueden evaluar los proyectos de aula.

Para evaluar un proyecto de aula:

- Seleccionar la convocatoria, el evaluador y el proyectos asignado.
- Seleccionar el parámetro a evaluar, ingresar la nota correspondiente y dar clic en el botón “Calificar Parámetro”. Repetir este procedimiento para los demás parámetros.
- Cuando se evalúen todos los parámetros dar clic en el botón “Calificar Proyecto”.

Descripción	Valor Máximo	Nota	Nota Final
PRUEBA	10	0	0

Figura F.40. Pantalla Evaluar Semillero.
Fuente: [Propia].

3. Aprobar Semillero

En este formulario podemos aprobar los proyectos de aula que previamente tienen que ser evaluados por los evaluadores.

Para aprobar la evaluación del proyecto de aula:

- Identificar la convocatoria en la que está registrado el proyecto de aula.
- Seleccionar el proyecto que a aprobar.
- Presionar el botón “Aprobar Proyecto”, si el puntaje obtenido por el proyecto es el requerido el proyecto aprueba, caso contrario este no aprueba.

The screenshot shows a web application window titled 'APROBAR SEMILLEROS'. The interface is divided into several sections:

- CONVOCATORIA:** Includes fields for 'Estado' (checkbox), 'Tipo Convocatoria' (dropdown), 'Periodo Academico', 'Tema', 'Fecha Convocatoria', and 'Formato Proyecto'.
- SEMILLERO:** Includes fields for 'Estado' and 'Código'.
- Título:** A large text input field.
- PUNTAJE EVALUACION:** A table with two columns: 'Parámetro' and 'Promedio'. It contains four rows for data entry and a 'Total' row at the bottom.
- ACCIONES:** A section containing a single button labeled 'Aprobar Proyecto'.

Figura F.41. Pantalla Aprobar Semillero.
Fuente: [Propia].

4. Semilleros Triunfadores

En este formulario podemos ver los proyectos de aula triunfadores, luego de haber sido aprobados por las respectivas autoridades.

Para ver los proyectos de aula triunfadores:

- Identificar la convocatoria en la que está registrado el proyecto de aula.
- Seleccionar el proyecto de aula.
- Verificar el porcentaje obtenido en su evaluación y verificar el estado obtenido.
- Para revisar un proyecto específico basta con hacer una búsqueda personalizada haciendo clic en el botón “Buscar” e ingresar el criterio de búsqueda.

Figura F.42. Pantalla Aprobar Semillero.
Fuente: [Propia].

➤ Eventos

1. Administrar Evento

En esta sección podemos crear un evento (seminario, taller, capacitación, etc.) como resultado de un proyecto de investigación. Para crear un evento hacemos lo siguiente:

- Seleccionar la convocatoria realizada para el evento a programar.
- Insertar un registro dando clic en el botón “Insertar”.
- Llenar los datos requeridos en el formulario.
- Guardar la información registrada dando clic en el botón “Guardar”.

Figura F.43. Pantalla Administrar Evento.
Fuente: [Propia].

2. Facilitadores del Evento

Por medio de este formulario podemos ingresar los facilitadores del evento y para eso tenemos que realizar lo siguiente:

- Identificar el evento del cual se quiere ingresar el o los facilitadores.
- Seleccionar el tipo de facilitador (investigador, docente u otro).
- Ubicarse en la sección “Facilitador” y dar clic en el botón “Insertar”.
- Dar clic en el botón “Listar” para mostrar la lista de los posibles facilitadores del evento, esta lista será mostrada de acuerdo al tipo de facilitador seleccionado.
- Ingresar la información requerida en el formulario: cédula, nombre, estado, descripción y observación en el caso de que se quiera poner.
- Para guardar damos clic en el botón “Guardar”.

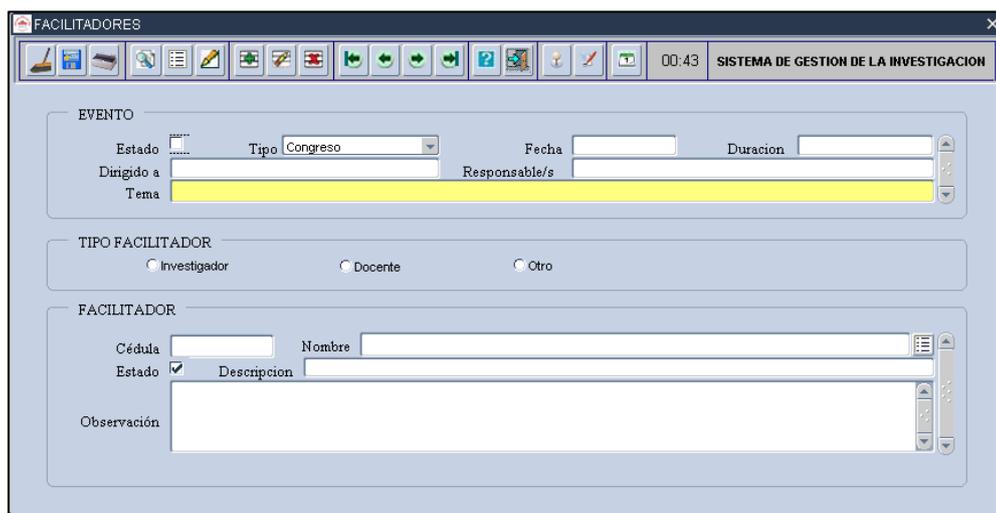


Figura F.44. Pantalla Facilitadores Evento.
Fuente: [Propia].

3. Asistentes Evento

En este formulario podemos registrar a los asistentes del evento realizado, para lo cual hacemos lo siguiente:

- Identificar el evento del cual se quiere ingresar el o los asistentes.
- Seleccionar el tipo de asistente (investigador, docente u otro).
- Ubicarse en la sección “Asistente” y dar clic en el botón “Insertar”.

- Dar clic en el botón “Listar” para mostrar la lista de los posibles asistentes del evento, esta lista será mostrada de acuerdo al tipo de asistente seleccionado.
- Ingresar la información requerida en el formulario: cédula, nombre, estado, descripción y observación en el caso de que se quiera poner.
- Para guardar damos clic en el botón “Guardar”.

Figura F.45.Pantalla Asistentes Evento.
Fuente: [Propia].

4. Contenido Evento

En este formulario podemos registrar el contenido del evento realizado, para lo cual hacemos lo siguiente:

- Seleccionar el evento.
- Ubicar el cursor del ratón en la sección “Planificación”.
- Para insertar el registro damos clic en el botón “Insertar”.
- Para asignar el facilitador responsable del ítem que se está ingresando hay que levantar la lista de valores dando clic en el botón “Listar”.
- Ingresamos la información requerida: estado, nro. , contenido, responsable, facilitador, duración y fecha.
- Para guardar damos clic en el botón “Guardar”.

- Si se quiere seguir insertando más ítems, hay que seguir la secuencia de pasos detallados con anterioridad.

Figura F.46. Pantalla Contenido Evento.
Fuente: [Propia].

➤ Publicaciones

1. Artículos Científicos

En este formulario podemos ver los artículos científicos publicados con la ejecución de los proyectos de investigación, para lo cual tenemos que realizar lo siguiente:

- Seleccionar el nombre del proyecto.
- Navegar por el bloque de datos mostrado.
- Buscar una publicación de un artículo científico con un criterio de búsqueda.

Figura F.47. Pantalla Artículos Científicos.
Fuente: [Propia].

2. Producción Científica

En este formulario podemos ver la producción científica generada con la ejecución de los proyectos de investigación, para lo cual tenemos que realizar lo siguiente:

- Seleccionar el nombre del proyecto.
- Navegar por el bloque de datos mostrado.
- Buscar información de acuerdo a un criterio de búsqueda.

The screenshot shows a software window titled 'PRODUCCIÓN CIENTÍFICA' with a toolbar and a status bar. The main area is divided into two sections: 'PROYECTO' and 'PRODUCCIÓN CIENTÍFICA'. In the 'PROYECTO' section, the 'Estado' is 'Formulación', the 'Director' is '1002641288 MEJIA ECHEVERRIA COSME DAMIAN', and the 'Titulo' is 'Sistema semiautomático de mezclado de leche y corte de la cuajada para una cuba quesera de 500 litros en las unidades productivas de la FICAYA'. The 'PRODUCCIÓN CIENTÍFICA' section contains a table with the following columns: 'Titulo', 'Tipo', 'Fecha Inicio', 'Estado', and 'Fecha Terminación'. The table has one row with 'Institucionales' in the 'Tipo' column and 'En Ejecución' in the 'Estado' column.

Titulo	Tipo	Fecha Inicio	Estado	Fecha Terminación
	Institucionales		En Ejecución	

Figura F.48. Pantalla Producción Científica.

Fuente: [Propia].

F.2.3. Sección Procesos

En la sección de procesos se despliegan todos los procesos principales que necesitamos para poner en funcionamiento el resto del sistema.

➤ Proyectos

1. Presentación

A través de este formulario se hace el ingreso de los proyectos de investigación, para lo cual realizamos lo siguiente:

- En el menú escogemos: procesos, proyectos, presentación e Ingresar Proyecto.
- Seleccionamos la convocatoria en la que se quiere registrar el proyecto.
- Para insertar un nuevo registro damos pulsamos el botón "Insertar Registro" e ingresamos los datos solicitados en el formulario.

- Para guardar damos clic en el botón “Guardar”.
- Hay que tener en cuenta que el formulario tiene pestañas como: Datos Relevantes, Equipo de trabajo y Descripción; para esto realizamos los pasos anteriores.

The screenshot shows a web application window titled 'INGRESAR PROYECTO' with a toolbar and a status bar. The main content area has four tabs: 'Datos Generales', 'Datos Relevantes', 'Equipo Trabajo', and 'Descripción'. The 'Datos Relevantes' tab is active. Under 'CONVOCATORIA', there are fields for 'Estado' (checked), 'Tipo' (Proyectos Internos), 'Periodo' (2011-2012), 'Fecha' (10/02/2012), and 'Formato' (CUICYT). The 'Tema' field contains 'Convocatoria a presentación de proyectos internos 2012'. Under 'PROYECTO', there are fields for 'Código' (000000001), 'Definición' (Proyecto), 'Tipología' (Investigación), 'Categoria' (Investigador), 'Título Corto' (Determinación y Selección de Bacterias Solubilizadoras de Potasio), and 'Título Completo' (Determinación y Selección de Bacterias Solubilizadoras de Potasio, provenientes de muestras de suelo de la Granja Experimental "Yuyucocha", como Fase Inicial para establecer un Biofertilizante de Potasio). Other fields include 'Duración (meses)' (6), 'Tipo' (Nuevo), 'Estado' (Formulación), 'Cobertura' (Provincial), 'Localización' (IMBABURA - ECUADOR), 'Institución Ejecutora' (Universidad Técnica del Norte), 'Dependencia' (FACULTAD DE INGENIERIA CIENCIAS AGROPECUARIAS Y AMBIENTALES - UNIVERSIDAD TECNICA DEL NORTE), 'Director' (GUEVARA GRANJA MARIA FERNANDA), 'Fecha Inicio' (01/01/2013), and 'Fecha Finalización' (28/07/2013). There are also fields for 'Monto Requerido', 'Monto Aprobado', 'Monto Gastado', and 'Monto Disponible', all set to 0. A 'Justificación' field contains text about agricultural productivity and soil quality. A 'Cargar Documento' button is at the bottom.

Figura F.49. Pantalla Ingresar Proyecto.
Fuente: [Propia].

2. Evaluación

Por medio de este formulario el evaluador asignado puede evaluar los proyectos de investigación, para lo cual debe realizar lo siguiente:

- En el menú escogemos: procesos, proyectos, evaluación y Evaluar Proyecto
- Seleccionar la convocatoria.
- Seleccionar el proyecto.
- Seleccionar el parámetro de evaluación.
- Ingresar la calificación respectiva.
- Guardar dando clic en el botón “Calificar Parámetro”.
- Registrar las observaciones, en el caso de que las hubiere, dando clic en el botón “Observaciones”.
- Repetir el mismo procedimiento para calificar el resto de parámetros.
- Una vez calificados todos los parámetros, procedemos a calificar el proyecto, dando clic en el botón “Calificar Proyecto”.

The screenshot shows the 'EVALUACION PROYECTOS' interface. It includes fields for 'CONVOCATORIA' (Estado, Período Académico: 2011-2012, Tema: Convocatoria a presentación de proyectos internos 2012, Tipo Convocatoria: Proyectos Internos, Fecha Convocatoria: 10/02/2012, Formato Proyecto: CUICT), 'EVALUADOR' (Cédula: 1001506052, Nombre: JUAN CARLOS GARCIA PINCHAO), 'PROYECTO' (Código: 000000019, Tema: GESTION DE LA INVESTIGACION, Tipo: Proyecto, Director: JOSE FERNANDO GARRIDO SANCHEZ), and 'PARAMETROS DE EVALUACION' (Descripción: CALIDAD CIENTIFICA, Valor Máximo: 70). A table for 'NOTA' shows scores for OBJETIVOS (15/25), METODOLOGIA (25/25), and PERTINENCIA (19/20). 'ACCIONES' includes buttons for 'Calificar Parametro', 'Calificar Proyecto', and 'Observaciones'.

Figura F.50. Pantalla Evaluar Proyecto.

Fuente: [Propia].

3. Aprobación

A través de este formulario se aprueban los proyectos de investigación, para lo cual realizamos lo siguiente:

- En el menú escogemos: procesos, proyectos, aprobación y Aprobar Proyecto.
- Seleccionamos la convocatoria.
- Seleccionamos el proyecto.
- Desplegamos la lista de valores presionando las teclas “Ctrl + L”, para anexar la resolución emitida para el proyecto.
- Para guardar los cambios presionamos el botón “Aprobar Proyecto”.

The screenshot shows the 'APROBAR PROYECTOS' interface. It includes fields for 'CONVOCATORIA' (Estado, Período Académico: 2011-2012, Tema: Convocatoria a presentación de proyectos internos 2012, Tipo Convocatoria: Proyectos Internos, Fecha Convocatoria: 10/02/2012, Formato Proyecto: CUICT), 'PROYECTO' (Estado: Revisado, Código: 000000007, Título: Entorno Dinámico de Aprendizaje para Personas con Capacidades Diferentes, Resolución: [empty]), and 'PUNTAJE EVALUACION' (table with columns 'Parámetro' and 'Promedio', and a 'Total' row). 'ACCIONES' includes a button for 'Aprobar Proyecto'.

Figura F.51. Pantalla Aprobar Proyecto.

Fuente: [Propia].

4. Oficialización

Por medio de este formulario se puede oficializar (cargar al POA) los proyectos de investigación, para lo cual debe realizar lo siguiente:

- En el menú escogemos: procesos, proyectos, oficialización y Cargar Proyectos.
- Seleccionar la convocatoria.
- Seleccionar el proyecto.
- Presionar el botón “Cargar” o “Cargar Todos”.
- Seleccionar la pestaña Actividades.
- Seleccionar la o las actividades a cargar.
- Cargar las actividades presionando el botón “Cargar” o “Cargar Todos”.
- Para quitar los proyectos y actividades cargadas utilizamos los botones “Quitar” o “Quitar Todos”.

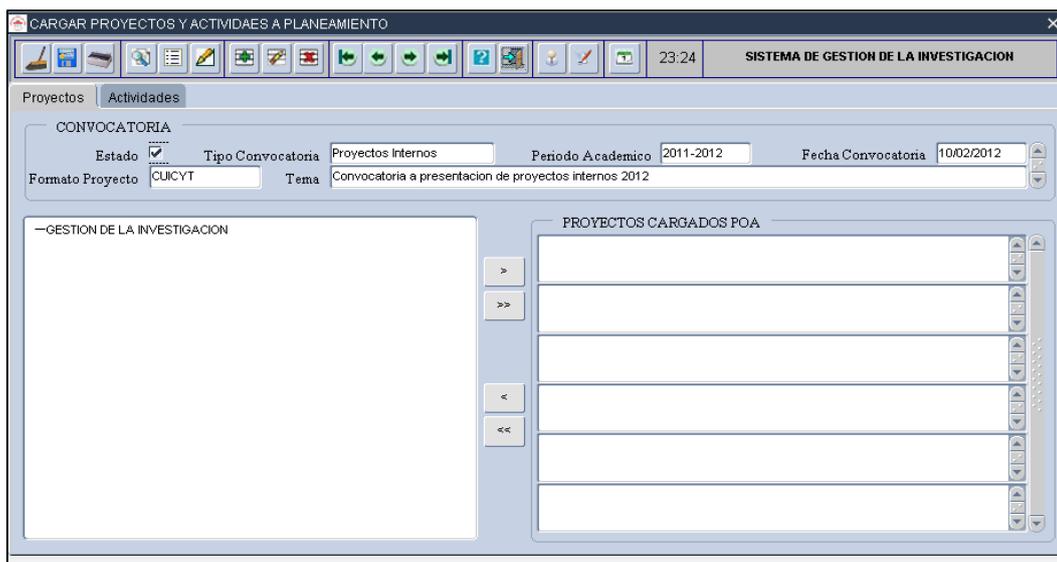


Figura F.52. Pantalla Oficializar Proyecto.

Fuente: [Propia].

5. Ejecución

En este formulario se crea informes de avance de proyectos y realizamos lo siguiente:

- En el menú escogemos: procesos, proyectos, ejecución e Informes.
- Seleccionamos la convocatoria y el proyecto de investigación para crear el informe.
- Presionamos el botón “Crear Informe”.

- Ingresamos los datos del informe y Guardamos dando clic en el botón “Guardar”.
- Ingresamos la información requerida en las pestañas del formulario y vamos guardando.

Actividad	% Parcial	% Acumulado	Monto Presupuestado	Monto Acum

Figura F.53. Pantalla Ejecutar Proyecto.
Fuente: [Propia].

F.2.4. Sección Reportes

En la sección de reportes se ejecutan los informes en formato PDF con los datos necesarios para validar los procesos de investigación.

➤ Proyectos por Periodo

Con este reporte podemos obtener una lista de los proyectos ejecutados en un período dado, para lo cual se debe seleccionar la definición de la investigación, seleccionar el período y presionar el botón “Ejecutar Reporte”.

Impresion

Direccion Numero de Pagina Institucion Sede
Portal Web Logo Usuario Telefono

PROYECTOS POR PERIODO

Definicion Investigacion [Dropdown]
Periodo [Dropdown]

Ejecutar Reporte

Figura F.54. Pantalla Proyecto por Periodo.
Fuente: [Propia].

➤ **Proyectos por Área de Investigación**

Con este reporte podemos obtener una lista de los proyectos por área de investigación, para lo cual se debe seleccionar el tipo de formato, seleccionar el área de investigación y presionar el botón “Ejecutar Reporte”.

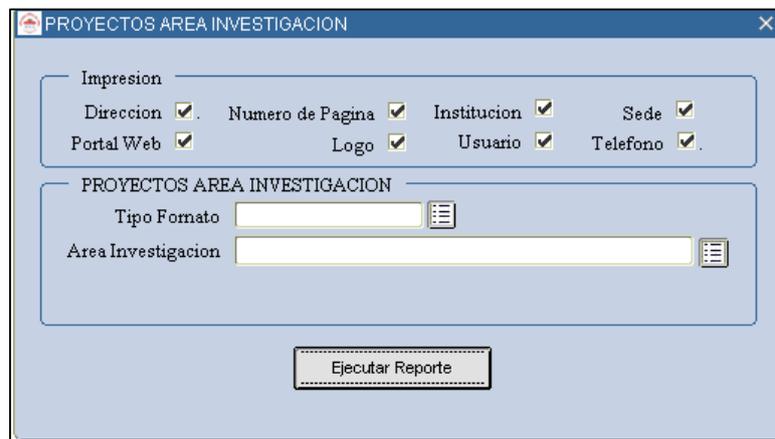


Figura F.55. Pantalla Proyecto por Área de Investigación.
Fuente: [Propia].

➤ **Proyecto Avance Técnico Económico**

Con este reporte podemos obtener un detalle con el avance técnico económico del proyecto en una convocatoria dada para lo cual se debe seleccionar la convocatoria, el proyecto, el número de informe y posteriormente presionar el botón “Ejecutar Reporte”.



Figura F.56. Pantalla Proyecto Avance Técnico Económico.
Fuente: [Propia].

G. REGLAMENTO INTERNO DEL CUICYT

REGLAMENTO DEL CENTRO UNIVERSITARIO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA – CUICYT

CAPÍTULO I DE LA CONSTITUCIÓN Y ORGANIZACIÓN

Art. 1.- El Centro Universitario de Investigación Científica y Tecnológica (CUICYT) depende del Vicerrectorado, de acuerdo con el Art. 69 del Estatuto Orgánico y es el organismo encargado de promover, coordinar y evaluar los proyectos de investigación del CUICYT y supervisar los Convenios Interinstitucionales.

Art. 2.- Las políticas de la dependencia se regirán a los lineamientos generales, objetivos y líneas prioritarias para la investigación científica y tecnológica de la Universidad Ecuatoriana por el Consejo Nacional de Universidades y Escuelas Politécnicas (CONUEP) y por los lineamientos aprobados por el H. Consejo Universitario.

Art. 3.- Los proyectos de investigación que no se encuentran contemplados dentro de las líneas prioritarias de investigación aprobados por el CONUEP, serán impulsados por el CUICYT buscando otras fuentes de financiamiento o con recursos propios que la universidad mantiene para investigación y compromiso de Convenios interinstitucionales.

Art. 4.- El Centro Universitario de Investigación Científica y Tecnológica "CUICYT" contará con los siguientes organismos, autoridades y personal de investigación y administrativo:

- a) El Consejo Directivo del CUICYT
- b) El Director
- c) El Coordinador de Proyectos CUICYT
- d) Los investigadores
- e) Apoyo administrativo

Además contará desde las Facultades, con el apoyo de los responsables de convenios profesores-investigadores y los Coordinadores de investigación de acuerdo con los Art. 139 y 140 del Reglamento General^[N1].

CAPÍTULO II DEL CONSEJO DIRECTIVO

Art. 5.- Se integrará de conformidad a lo que establece el Art. 69 del Estatuto Orgánico y es el organismo encargado de orientar y regular los proyectos de investigación del CUICYT, bajo las directrices del H. Consejo Universitario.

Art. 6.- Son sus funciones, a más de las contempladas en el Estatuto Orgánico las siguientes:

- a) Asesorar al H. Consejo Universitario y autoridades en todo cuanto se relacione con los proyectos de investigación del CUICYT.
- b) Aprobar los proyectos y evaluar los proyectos de investigación que genere la Universidad y los propuestos por organismos extra institucionales.
- c) Calificar, supervisar y evaluar los proyectos de investigación. Si el monto supera los 100 salarios mínimos vitales necesariamente se buscará financiamiento externo a la universidad; caso contrario se los financiará con recursos existentes en la partida correspondiente.
- d) Promover la formación de investigación y realización de eventos científicos-tecnológicos.
- e) Designar al Director del Proyecto de investigación, de entre los profesores-investigadores.
- f) Procurar el logro de fuentes de financiamiento para la investigación.
- g) Seleccionar el personal de investigación de los proyectos del CUICYT y de los convenios específicos de las Facultades y Universidad a petición del Director de Proyecto o responsable del Convenio.

^[N1]Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "De la Constitución y Organización", Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

- h) Suspender o dar por terminado los proyectos de investigación, cuando estos presenten novedades o problemas insalvables.
- i) Solicitar al H. Consejo Universitario, sanciones en caso de incumplimiento en la ejecución de proyectos de investigación.
- j) Proponer proyectos de reglamento, sobre las actividades de investigación.
- k) Fijar las condiciones a que se sujetarán los investigadores, cuando un proyecto sea financiado con recursos de la UTN.
- l) Seleccionar y solicitar contratos para asesores de evaluación técnica de los proyectos.
- m) Las demás que san asignadas por el H. Consejo Universitario^[N2].

CAPÍTULO III DEL DIRECTOR

Art. 7.- El Director del Centro de Investigación Científica y Tecnológica, será designado por el H. Consejo Universitario a propuesta en terna enviada, Vicerrector y laborará 40 horas semanales. En caso de remoción se estará a lo dispuesto en el Art. 17 numeral 21 del Estatuto.

Art. 8.- Son sus funciones:

- a) Cumplir y hacer cumplir, a su nivel, las disposiciones legales y reglamentarias.
- b) Representar oficialmente al "CUICYT" y responsabilizarse por su funcionamiento.
- c) Presentar el plan anual de trabajo hasta el mes de julio de cada año ante el Consejo Directivo, así como el Informe de labores.
- d) Elaborar el proyecto de presupuesto de su dependencia para someterlo a consideración del Consejo Directivo.
- e) Emitir informes que le soliciten el Consejo Directivo.
- f) Realizar la evaluación y el seguimiento periódico de los proyectos de investigación, informar al Consejo Directivo.
- g) Mantener relaciones con entidades afines tanto nacionales como extranjeras promoviendo la firma de contratos y convenios para la ejecución de proyectos de investigación y otros.
- h) Buscar financiamiento para los proyectos de investigación.
- i) Solicitar al Rector los contratos del personal técnico y administrativo de conformidad con los proyectos de investigación y con cargo a los recursos del mismo, o según los compromisos de Convenios.
- j) Distribuir y/o delegar funciones de responsabilidad entre el personal de su dependencia.
- k) Amonestar y tramitar sanciones del personal administrativo a su cargo argumentando sus razones.
- l) Promover y organizar la realización de cursos, encuentros, seminarios, talleres, etc., con otras entidades nacionales e internacionales afines, con la finalidad de intercambiar experiencias.
- m) Asesorar el trabajo de las Coordinaciones de Investigación de las Facultades.
- n) Otras que le asignen los Organismos y Autoridades Superiores^[N3].

CAPÍTULO IV DEL COORDINADOR DE PROYECTOS DEL CUICYT

Art. 9.- El Coordinador de Proyectos del CUICYT, deberá tener formación universitaria, será nombrado por el Rector previa solicitud del Consejo Directivo del CUICYT, siguiendo el trámite legal respectivo; laborará 40 horas semanales, y, tendrá las siguientes funciones:

- a) Representar oficialmente al CUICYT por delegación del Director.
- b) Coordinar las actividades con los Directores de Proyectos administrativos directamente por el CUICYT.

^[N2] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "Del Consejo Directivo", Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N3] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "Del Director", Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

- c) Tramitar a través del Director, los asuntos administrativos, desembolsos y requerimientos de los proyectos procurando que el personal técnico dedique la mayor parte del tiempo a los asuntos estrictamente científicos y tecnológicos.
- d) Actualizar permanentemente en cuanto a formulación y presentación de proyectos con el fin de asesorar a los profesores-investigadores.
- e) Velar por la continuidad y secuencia de las investigaciones y colaborar en la elaboración del plan anual y presupuesto del CUICYT.
- f) Colaborar con el avance y seguimiento de los proyectos de acuerdo con el cronograma respectivo e informar sobre los mismos.
- g) Revisar revistas y otros documentos científicos con el fin de proveer de información actualizada a los Directores de Proyectos.
- h) Colaborar en la ejecución de cursos, seminarios y otros eventos que organice el CUICYT.
- i) Llevar un listado de los profesores a dedicación exclusiva, tiempo completo y medio tiempo; y, controlar los plazos establecidos en el Reglamento General, comunicando las novedades al Director del CUICYT.
- j) Buscar los vínculos más idóneos de la investigación con la docencia y extensión universitaria.
- k) Responsabilizarse de los puntos nodales y focales que asuma la institución.
- l) Otras que le asignen los organismos y autoridades superiores^[N4].

CAPÍTULO V DE LOS INVESTIGADORES

Art. 10.- Los investigadores de la Universidad Técnica del Norte se ubicarán en tres grupos principales: de planta, adjuntos y contratados.

SERÁN INVESTIGADORES DE PLANTA de la Universidad Técnica del Norte. Los profesores-investigadores que dediquen sus actividades a la investigación y que participen en proyectos aprobados por la Institución. Sus actividades como profesores estarán normadas dentro de lo que establece el Estatuto Orgánico, Reglamento General y los Reglamentos de las Facultades.

SERÁN INVESTIGADORES ADJUNTOS.- Aquellos profesionales que posean las mismas condiciones que las requeridas para el personal de la Universidad Técnica del Norte y provengan de entidades con las cuales la UTN ha firmado un convenio para la ejecución de un proyecto. La Universidad Técnica del Norte no mantendrá ninguna relación de tipo laboral con los investigadores adjuntos, pudiendo esto si contribuir a su sostenimiento dentro de los términos del Convenio y por la duración del mismo. Los investigadores adjuntos podrán utilizar las facilidades logísticas necesarias para la ejecución del proyecto. Así como participar en las actividades de tipo cultural, social y recreacional. La Universidad Técnica del Norte podrá así mismo requerir del investigador adjunto la participación en actividades de tipo científico relativas a la difusión del proyecto.

LOS INVESTIGADORES CONTRATADOS.- Se denominarán Investigadores de Proyecto.

Los investigadores de Proyecto serán contratados exclusivamente para las funciones específicas que le determinen dentro del proyecto. Su relación contractual con la Universidad Técnica del Norte terminará al culminar el proyecto. Las condiciones y categorías están dadas en el Reglamento expedido por el CONESUP o por el Consejo Directivo del CUICYT, según el caso.

Art. 11.- Los profesores Universitarios a tiempo exclusivo, tiempo completo y medio tiempo tienen la obligación de presentar proyectos de investigación al Director del CUICYT, quien a la vez enviará al Consejo Directivo para su aprobación y el H. Consejo Directivo solicitará al H. Consejo Universitario se le designe como profesor-investigador. Se observará también en este proceso lo que establecen los Arts. 158 y 159 del Reglamento General.

Art. 12.- Los profesores investigadores contemplan su carga horaria desarrollando el proyectos de investigación. Si necesita recurrir a otros lugares para realizar investigación deberá comunicarlo al Director del CUICYT.

Art. 13.- Los docentes que alcance la denominación de profesor-investigador, recibirán un estímulo económico que fijará el H. Consejo Universitario y se hará efectivo una sola vez, al obtener la aprobación del proyecto.

^[N4] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "Del Coordinador de Proyectos del CUICYT", Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

Art. 14.- Además del sueldo y bonificaciones correspondientes los profesores-investigadores, según su denominación contemplada en el proyecto, recibirán las cantidades fijadas por el CONESUP, para estos casos. Se exceptúe de este procedimiento los profesores-investigadores que ejecuten proyectos aprobados por el CUICYT, por el bajo costo de la investigación.

Art.15.- Los profesores-investigadores presentarán informes en forma trimestral al Director del CUICYT, sobre el avance del proyecto que desarrolla.

Art. 16.- Para apoyo de las investigaciones el CUICYT, contará con auxiliares de investigación que apruebe el H. Consejo Universitario.

Art. 17.- Son atribuciones de los Auxiliares de Investigación:

- a) Colaborar con los profesores universitarios en la formulación del anteproyecto de investigación.
- b) Realizar investigaciones menores que le soliciten las autoridades superiores.
- c) Colaborar con los estudiantes y egresados en la realización de trabajos de investigación.
- d) Recopilar información cualitativa y cuantitativa en las diversas áreas del conocimiento humano.
- e) Colaborar en la elaboración del plan anual de actividades.
- f) Constatar el avance de la investigación de acuerdo a los cronogramas respectivos.
- g) Colaborar previa la autorización del Director, en proyectos de investigación.
- h) Colaborar en la elaboración del presupuesto de la dependencia.
- i) Colaborar en la planificación y ejecución de cursos, seminarios y más eventos que organice la dependencia.
- j) Colaborar en la evaluación y seguimiento de los proyectos.
- k) Apoyar a los programas de Extensión Universitaria.
- l) Colaborar en la consecución de fondos para financiar los proyectos de investigación.
- m) Otras que le asigne el Director del CUICYT, Autoridades y Organismos Superiores^[N5].

CAPÍTULO VI DEL APOYO ADMINISTRATIVO

Art. 18.- El Centro Universitario de Investigación Científica y Tecnológica "CUICYT", contará con el personal administrativo y de servicio, que apruebe el H. Consejo Universitario.

Art. 19.- Las funciones del personal administrativo constará en el Manual y valoración de puestos de la UTN^[N6].

CAPÍTULO VII NORMAS DE PROCEDIMIENTOS DE LA INVESTIGACIÓN

Art. 20. Las actividades que desarrolle la UTN serán analizadas a través de PROYECTOS, cuyos lineamientos consten en un documento escrito. Estos documentos serán presentados en un formato establecido y servirán de base para análisis de los proyectos.

Art. 21.- La investigación específica planteada en cada proyecto será ejecutado por uno o más investigadores. Habrá un Director en cada proyecto, el cual será responsable por la ejecución de los mismos. Se presentarán informes trimestrales y un informe final que servirán para la evaluación del proyecto en ejecución. Las evaluaciones de los proyectos serán hechas por el Consejo Directivo, el que podrá consultar las opiniones que considere apropiadas. Para ser Director de un Proyecto se requiere ser profesor de la UTN. La investigación podrá desarrollarse tanto a nivel personal por un investigador o por grupo de investigadores.

^[N5] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "De los Investigadores" Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N6] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "Del Apoyo Administrativo" Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

Art. 22.- La propiedad intelectual de los resultados de la investigación que se desarrolla en la UTN será patrimonio de ésta, y así deberá indicarse en cada propuesta de proyecto. En caso de participar otras instituciones, se dará el crédito y reconocimiento apropiados a las mismas.

Art. 23.- La parte obtenida en base a trabajos de la UTN, será de propiedad exclusiva de la institución.

Art. 24.- Las actividades de investigación que se desarrollen en la UTN deberán encuadrarse claramente dentro de los objetivos y finalidad institucionales, de acuerdo a los lineamientos de la filosofía de la investigación.

Art. 25.- El H. Consejo Universitario pondrá límites de los emolumentos adicionales que podrán recibir los investigadores que se encuentran trabajando en proyectos de investigación.

Estos emolumentos adicionales deberán estar establecidos en el esquema de costos del Proyecto y aprobados por el Consejo Directivo, debiendo provenir de fuentes externas. Por otro lado, y para premiar actividades destacadas en el ámbito de la investigación, el Consejo Directivo establecerá presea que conlleven un componente pecuniario del monto aprobado, el que se entregará estrictamente en base a mérito demostrado y evaluados de acuerdo a un reglamento especial^[N7].

CAPÍTULO VIII PROYECTOS DE INVESTIGACIÓN

Art. 26.- Denominase Proyecto de Investigación al conjunto de actividades de tipo científico y tecnológico que conducen el desarrollo de una investigación específica. Para la aprobación de un proyecto por el Consejo Directivo, se deberá presentar una propuesta del proyecto en el formato establecido. Una vez aprobados, se considera proyecto en ejecución.

Para ser aprobado un proyecto deberá establecer claramente el objetivo del proyecto, su justificación, beneficios e impactos, costos, personal partícipe, cronograma de trabajo y fuentes posibles de financiación. Deberá así mismo presentar un análisis del personal y de la infraestructura técnica requerida, especificando existencias actuales, dentro de la UTN y requerimientos adicionales para la ejecución del proyecto, incluyendo claramente el sufragio de gastos de personal y de las actividades de investigación.

Art. 27.- El investigador principal o Directos de cada proyecto deberá presentar los informes correspondientes. El informe final deberá presentarse dentro de los cuatro meses siguientes a la finalización del plazo inicialmente fijado para el proyecto. En caso necesario, el Consejo Directivo del CUICYT considerará la extensión de prórroga para la finalización del proyecto, en cuyo caso deberá presentarse el informe final dentro de los cuatro meses de la nueva fecha de finalización^[N8].

CAPÍTULO IX DE LAS SANCIONES

Art. 28.- En caso de no presentar los informes o una justificación satisfactoria al Consejo Directivo del CUICYT, este recomendará al Rector alguna sanción disciplinaria.

Las sanciones disciplinarias que se tome dependerá de la magnitud del proyecto de los compromisos adquiridos, en especial con entidades externas. En caso que existan compromisos de tipo legal, especialmente en aquellos en que el investigador hubiese recibido asignaciones económicas específicas para la ejecución de un proyecto y éste se incumpliere, la UTN podrá tomar las acciones legales correspondientes.

Art. 29.- Cuando por negligencia del Director del Proyecto se suspenda la investigación de éste, deberá devolver a través del Director Financiero las bonificaciones contempladas en el Art. 13 del presente Reglamento. Dentro de los 30 días subsiguientes a la declaratoria de suspendido por parte del Consejo Directivo del CUICYT, caso contrario se procederá el descuento en el rol de pagos.

^[N7] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "**Normas de Procedimiento de Investigación**" Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

^[N8] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "**Proyectos de Investigación**" Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.

Art. 30.- Para no perder su dedicación de profesor-investigador, cuando se ha suspendido un proyecto, puede presentar otro proyecto de investigación de los subsiguientes 120 días.

Art. 31.- En caso de reincidencia será sancionado conforme establece el Art. 174 del Estatuto Orgánico numerales 3, 4 y 5.

Dado en la sala de Sesiones del Honorable Consejo Universitario a los veintisiete días del mes de noviembre de mil novecientos noventa y uno^[N9].

Dr. Fausto Álvarez
SECRETARIO GENERAL

^[N9] Reglamento Interno del Centro Universitario de Investigación, Ciencia y Tecnología (CUICYT). "De las **sanciones**" Honorable Consejo Universitario (HCU), 17 Noviembre de 1991, Universidad Técnica del Norte.