



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

**TESIS PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO
SISTEMAS E INFORMÁTICA**

**AUTORES: ALTAMIRANO BASTIDAS OSCAR MAURICIO
CARATE PACHECO OSCAR SALOMÓN**

**TEMA: ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE
LOS MÓDULOS DE ADMINISTRACIÓN, ACADÉMICO Y DOBE
ORIENTADO A LA WEB PARA EL COLEGIO MILITAR Nº 10 ABDÓN
CALDERÓN”**

**DIRECTOR: ING. CÓRDOVA RAÚL
CODIRECTOR: ING. MONTENEGRO CARLOS**

SANGOLQUÍ, JULIO 2015

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

CERTIFICADO

Ing. Raúl Córdova
Ing. Carlos Montenegro

CERTIFICAN

Que el trabajo titulado “ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE LOS MÓDULOS DE ADMINISTRACIÓN, ACADÉMICO Y DOBE ORIENTADO A LA WEB PARA EL COLEGIO MILITAR N° 10 ABDÓN CALDERÓN”, realizado en su totalidad por los Sres. Oscar Mauricio Altamirano Bastidas y Oscar Salomón Carate Pacheco ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la Universidad de las Fuerzas Armadas – ESPE.

El mencionado trabajo consta de un documento empastado y un disco compacto al cual contiene los archivos en formato portátil de Acrobat (PDF). Se autoriza a los Sres. Oscar Altamirano Bastidas y Oscar Carate Pacheco, que el material se entregue al Ing. Mauricio Campaña en su calidad de Director de Carrera.

Sangolquí, Julio de 2015.



Ing. Raúl Córdova

DIRECTOR



Ing. Carlos Montenegro

CODIRECTOR

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

DECLARACIÓN DE RESPONSABILIDAD

Altamirano Bastidas Oscar Mauricio
Carate Pacheco Oscar Salomón

DECLARAN QUE:

La presente tesis de grado titulada “ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE LOS MÓDULOS DE ADMINISTRACIÓN, ACADÉMICO Y DOBE ORIENTADO A LA WEB PARA EL COLEGIO MILITAR N° 10 ABDÓN CALDERÓN”, ha sido desarrollada en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Julio de 2015



Sr. Oscar Altamirano B.

CC: 1710452606



Sr. Oscar Carate P.

CC: 1715336689

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

AUTORIZACIÓN DE PUBLICACIÓN

Altamirano Bastidas Oscar Mauricio
Carate Pacheco Oscar Salomón

AUTORIZAN

A la Universidad de las Fuerzas Armadas – ESPE la publicación en la Biblioteca Virtual de la Institución del trabajo “ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE LOS MÓDULOS DE ADMINISTRACIÓN, ACADÉMICO Y DOBE ORIENTADO A LA WEB PARA EL COLEGIO MILITAR N° 10 ABDÓN CALDERÓN”, cuyo contenido, ideas y criterio son de nuestra responsabilidad y autoría.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Julio de 2015



Sr. Oscar Altamirano B.

CC: 1710452606



Sr. Oscar Carate P.

CC: 1715336689

DEDICATORIA

Dedico esta tesis a mi madre, hermana y mis tíos que siempre han estado conmigo apoyándome incondicionalmente en todo momento en las buenas y en las malas. Han sido un puntal extremadamente importante y alto a seguir en toda mi vida los quiero mucho.

Oscar Mauricio Altamirano Bastidas

DEDICATORIA

Esta tesis está dedicada a mis padres, hermanas y sobrino. Ellos son parte fundamental en mi vida. Gracias por su apoyo en todos los momentos. Son un gran pilar que me ayudan a seguir todos los días.

Oscar Salomón Carate Pacheco

AGRADECIMIENTOS

Dedico esta tesis a mi madre, hermana y mis tíos que siempre han estado conmigo apoyándome incondicionalmente en todo momento en las buenas y en las malas. Han sido un puntal extremadamente importante y alto a seguir en toda mi vida los quiero mucho.

Oscar Mauricio Altamirano Bastidas

AGRADECIMIENTOS

Agradezco a Dios por darme fuerzas a continuar día a día pese a todas las adversidades encontradas en el camino.

A mis padres Elvia y Edgar, a mis hermanas Ana y Tania, a mis abuelos, tíos, primos, amigos y compañeros por el apoyo que siempre he recibido por parte de ellos.

Agradezco además, a mi director de tesis, Ing. Raúl Córdova quien con su apoyo, conocimiento y sobretodo paciencia a contribuido a que pueda terminar este proyecto la tesis.

A la Universidad, la cual a través de sus docentes me formaron profesionalmente y de la cual he adquirido muchos conocimientos que los aplico diariamente en mi vida profesional.

Oscar Salomón Carate Pacheco

INDICE DE CONTENIDOS

CERTIFICADO.....	ii
DECLARACIÓN DE RESPONSABILIDAD	iii
AUTORIZACIÓN DE PUBLICACIÓN.....	iv
DEDICATORIA	v
DEDICATORIA	vi
AGRADECIMIENTOS.....	vii
AGRADECIMIENTOS.....	viii
INDICE DE CONTENIDOS.....	ix
LISTADO DE TABLAS.....	xiv
LISTADO DE CUADROS.....	xv
LISTADO DE FIGURAS.....	xvi
LISTADO DE ANEXOS.....	xx
RESUMEN	xxi
ABSTRACT.....	xxii
CAPÍTULO 1.....	1
1.1. Introducción	1
1.2. Planteamiento del problema	1
1.3. Justificación	3
1.4. Objetivos	3
1.4.1 Objetivo General	3
1.4.2 Objetivos Específicos	4
1.5 Alcance	4
1.6 Beneficios para el Comil 10	4
CAPÍTULO 2.....	6
Marco Teorico.....	6
2.1 Introducción a AUP (Proceso Unificado Ágil) (Ambler, 2006) (Flores Ervin, 2013)	6
2.1.1. AUP	6
2.1.2. Ciclo de vida del Proceso Unificado Ágil (AUP).....	7
2.1.3. Principios de AUP	7

2.1.4.	Fases	8
	Elaboración	8
	Construcción	9
	Transición.....	9
2.1.5.	Disciplinas.....	9
	Modelo	9
	Implementación	9
	Prueba	9
	Despliegue	10
	Administración de la Configuración	10
	Administración del Proyecto	10
	Ambiente	10
2.1.6.	Iteraciones	10
2.1.7.	Roles de equipo	11
2.1.8.	Entregables.....	12
2.2	Lenguaje Unificado de Modelado (UML).....	13
2.2.1.	Definición de UML.....	13
2.2.2	Diagramas UML	14
2.3.	Modelo tres capas de una Aplicación Web	17
2.4.	Seguridades en Aplicaciones Web	18
2.4.1	Medidas de prevención contra las vulnerabilidades	19
2.5.	Ventajas de una Aplicación Web	21
2.6.	Desventajas de una Aplicación Web (Alegsa, 2013)	22
2.7.	Herramientas, lenguajes de programación y servicios.....	23
2.7.1.	Microsoft Visual Studio Express Edition	23
2.7.2.	Microsoft SQL Server 2008 Express	23
2.7.3.	Internet Information Services o IIS	23
2.7.4.	JavaScript	24
2.7.5.	Lenguaje C#	24
2.7.6.	Transact-SQL (T-SQL)	24
2.7.7.	Windows Server 2003	24

CAPÍTULO 3.....	25
3.1. Fase de Inicio.....	25
3.1.1. Alcance.....	25
3.1.1.1. Alcance del Módulo de Administración.....	25
3.1.1.2. Alcance del Módulo Académico.....	25
3.1.1.3. Alcance del Módulo DOBE.....	26
3.1.2. Especificación de Requerimientos IEEE 830.....	26
3.1.2.1. Definiciones, Acrónimos y Abreviaturas.....	26
3.1.2.2. Definiciones.....	26
3.1.2.3. Acrónimo.....	27
3.1.2.4 ANEXO A: Especificación de Requerimientos según Norma IEEE 830.....	28
3.1.3. Especificación de Casos de Uso.....	29
3.1.3.1. ANEXO B: Especificación de Casos de Uso y Diagramas....	29
3.1.4. MODELOS DE DOMINIO.....	29
3.1.4.1. Sub Módulo Secretaria.....	31
3.1.4.2 Módulo DOBE.....	33
3.1.6. Definición de riesgos.....	35
3.1.7. Matriz de Interesados.....	36
3.1.8. ANEXO C: Diagrama Conceptual de Base de Datos.....	36
3.1.9. ANEXO D: Diagrama Físico de Base de Datos Página.....	37
3.2. Fase de Elaboración.....	37
3.2.1. Objetivo: obtener la arquitectura del sistema.....	37
3.2.2. Identificar arquitectura.....	37
3.2.2.1. Capa de Presentación:.....	37
3.2.2.2. Capa de Aplicación:.....	37
3.2.2.3. Capa de Lógica:.....	38
3.2.2.4. Capa de Acceso a Datos:.....	38
3.2.3 Vista Lógica.....	40
3.2.4.1. Estación cliente:.....	41
3.2.4.2. Servidor Web IIS y de Aplicación:.....	41
3.2.4.3. Servidor de Base de datos:.....	41

3.2.5.	Diseño de Interfaz Gráfica.....	42
3.2.5.1.	Estándar de Interfaz Gráfica	42
3.2.5.2.	Encabezado de página:	43
3.2.5.3.	Nombre del Colegio:	43
3.2.5.4.	Nombre de usuario:	43
3.2.5.5.	Perfil de usuario:	43
3.2.5.6.	Fecha de sesión:.....	43
3.2.5.7.	Barra de menú:	44
3.2.5.8.	Icono de desconexión:	44
3.2.5.9.	Barras de desplazamiento:	44
3.2.5.10.	Hojas de estilos en cascada (CSS):.....	44
3.2.6.	Restricciones Generales	44
3.2.6.1.	Políticas Reguladoras	44
3.2.6.2.	Limitaciones del Hardware	45
3.2.6.3.	Limitaciones de Software	45
3.2.6.4.	Funcionamiento en Paralelo	45
3.2.6.5.	Requisitos de Lenguaje	46
3.2.6.6.	Protocolos Señalados	46
3.2.6.7.	Requisitos de Fiabilidad.....	46
3.2.6.8.	Seguridad y Condiciones de Seguridad	46
3.2.7.	Consideraciones finales	47
3.3.	Fase de Construcción	47
3.3.1.	Estructura General	47
3.3.2.	Estándares de Desarrollo.....	48
3.3.3.	Estándares de Codificación	48
3.3.3.1.	Namespace.....	48
3.3.3.2.	Clases	48
3.3.3.3.	Métodos	50
3.3.3.4.	Variables	51
3.3.3.5.	Comentarios.....	51
3.3.3.6.	Formularios	51
3.3.3.7.	Archivos de validación Java Script.....	51

3.3.4.	Estándares de Base de Datos.....	52
3.3.4.1.	Tablas	52
3.3.4.2.	Campos tablas	52
3.3.4.3.	Procedimientos almacenados	53
3.3.4.4.	Funciones	53
3.3.5.	Desarrollo de interfaces de Usuario	53
3.3.5.1.	Diseño formulario de Usuario.....	53
3.3.5.2.	Diseño formulario de Estudiante	54
3.3.5.3.	Diseño formulario de Ficha Psicológica	55
3.3.5.4.	Diseño formulario de Estudiante Nota	55
3.4.	Fase de Transición	56
3.4.1.	Pruebas.....	56
3.4.1.1.	Pruebas de Unidad	56
3.4.1.2.	ANEXO E: Pruebas Unitarias de Software.....	57
3.4.2.	Pruebas de Stress.....	57
3.4.3.	Instalación del Sistema	59
3.4.3.1.	Puesta en Producción.....	60
3.4.3.2.	Compilación de la Solución.....	60
3.4.3.3.	Configuración y generación de código.....	60
3.4.3.4.	Configuración del Servidor de Producción(IIS)	61
3.4.3.5.	Creación del Sitio Web EWEB	61
3.4.3.6.	Asignación de la IP Pública.....	62
3.4.3.8.	Sitio Web en producción	63
3.4.3.9.	Pantalla inicial de sistema.....	63
3.4.3.10.	Búsqueda de estudiantes.....	64
3.4.3.11.	Creación de estudiantes	64
3.4.3.12.	Catálogo General.....	65
CAPÍTULO 4.....	66	
4.1.	Conclusiones y Recomendaciones	66
4.1.1.	Conclusiones.....	66
4.1.2.	Recomendaciones.....	67
	Bibliografía	68

LISTADO DE TABLAS

Tabla 3. 1 Definición de Riesgos	35
Tabla 3. 2 Matriz de Interesados.....	36
Tabla 3. 3 Comparación requerimientos no Funcionales y Solución Propuesta	39

LISTADO DE CUADROS

Cuadro 3. 1 Estimación Costos y Gastos	34
--	----

LISTADO DE FIGURAS

Figura 2. 1 AUP (Fases de Ciclo de vida).....	7
Figura 2. 2 Iteraciones para los Releases de Producción.....	11
Figura 2. 3 Arquitectura Web en Tres Capas.....	18
Figura 3. 1 Modelo de Dominio de Administración Módulo Académico	29
Figura 3. 2 Modelo de Dominio de Académico	30
Figura 3. 3 Modelo de Dominio Sub Módulo Secretaria 1	31
Figura 3. 4 Modelo de Dominio Sub Módulo Secretaria 2.....	32
Figura 3. 5 Modelo de Dominio DOBE.....	33
Figura 3. 6 Vista lógica del sistema	40
Figura 3. 7 Diagrama de despliegue	41
Figura 3. 8 Interfaz Gráfica Pagina Inicio.....	42
Figura 3. 9 Interfaz Gráfica Aplicación.....	43
Figura 3. 10 Estructura General de la Aplicación.....	47
Figura 3. 11 Ejemplo de Clases.....	48
Figura 3. 12 Ejemplo Clase Namespace.....	49
Figura 3. 13 Clases Módulo Administración.....	49
Figura 3. 14 Clases Sub Módulo Secretaria	50
Figura 3. 15 Clase en Namespace Secretaria	50
Figura 3. 16 Ejemplos de Métodos	51
Figura 3. 17 Ejemplos de Variables	51
Figura 3. 18 Ejemplo de Comentarios.....	51
Figura 3. 19 Estándar Nombre de formularios	51
Figura 3. 20 Estándar Nombre de formularios	51
Figura 3. 21 Estándar nombre Tablas.....	52
Figura 3. 22 Estándar campos de la tabla	52
Figura 3. 23 Estándar Campos de tabla	53
Figura 3. 24 Estándar Procedimientos Almacenados	53
Figura 3. 25 Estándar Funciones de Base de Datos.....	53
Figura 3. 26 Diseño del formulario de Usuario.....	54
Figura 3. 27 Diseño del formulario de Estudiante	55
Figura 3. 28 Diseño del formulario de Ficha Psicología.....	55
Figura 3. 29 Diseño del formulario de Estudiante Nota.....	56
Figura 3. 30 Módulos de Prueba.....	57
Figura 3. 31 Configuración pruebas de stress	58
Figura 3. 32 Resultado Pruebas de Stress	58
Figura 3. 33 Resultado pruebas de Stress Usuarios.....	59
Figura 3. 34 Resultado pruebas de Stress Usuarios.....	59
Figura 3. 35 Publicar Solución	60
Figura 3. 36 Configurar Carpeta Solución.....	60
Figura 3. 37 Configurar Servidor Web IIS	61

Figura 3. 38 Crear Nombre Sitio Web.....	61
Figura 3. 39 Crear Nombre Sitio Web.....	61
Figura 3. 40 Asignar IP Pública Sitio Web	62
Figura 3. 41 Asignar Permisos Sitio Web	62
Figura 3. 42 Vista Sitio Web.....	63
Figura 3. 43 Formulario Ingreso a la Aplicación.....	63
Figura 3. 44 Formulario de Búsqueda de Estudiantes	64
Figura 3. 45 Formulario Estudiante.....	64
Figura 3. 46 Catálogo General.....	65
Figura B. 1 Paquetes Módulos de la Aplicación.....	94
Figura B. 2 Paquetes Módulo Administración	95
Figura B. 3 Caso de Uso Gestionar Perfil	98
Figura B. 4 Caso de Uso Asignar Perfil Opción	99
Figura B. 5 Caso de Uso Gestionar Usuario.....	102
Figura B. 6 Caso de Uso Primer Ingreso	105
Figura B. 7 Caso de Uso Inactivar Usuario.....	106
Figura B. 8 Paquetes Módulo Académico.....	107
Figura B. 9 Paquetes Sub Módulo Secretaria	108
Figura B. 10 Caso de Uso Gestionar Estudiante	111
Figura B. 11 Caso de Uso Gestionar Personal	114
Figura B. 12 Caso de Uso Gestionar Estado Civil	117
Figura B. 13 Caso de Uso Gestionar Raza.....	119
Figura B. 14 Caso de Uso Gestionar Religión	122
Figura B. 15 Caso de Uso Gestionar Tipo Sangre.....	124
Figura B. 16 Caso de Uso Gestionar País	127
Figura B. 17 Caso de Uso Gestionar Provincia	129
Figura B. 18 Caso de Uso Gestionar Cantón.....	131
Figura B. 19 Caso de Uso Gestionar Parroquia.....	134
Figura B. 20 Caso de Uso Gestionar Barrio.....	136
Figura B. 21 Caso de Uso Gestionar Familiar	139
Figura B. 22 Caso de Uso Gestionar Parentesco	142
Figura B. 23 Caso de Uso Gestión Idioma.....	144
Figura B. 24 Caso de Uso Gestión Instrucción	147
Figura B. 25 Caso de Uso Gestionar Año Lectivo.....	148
Figura B. 26 Caso de Uso Asignar Curso Paralelo	149
Figura B. 27 Caso de Uso Gestionar Curso Paralelo.....	150
Figura B. 28 Caso de Uso Asignar Estudiante Año Lectivo	152
Figura B. 29 Caso de Uso Generar Horario Clase.....	153
Figura B. 30 Caso de Uso Asignar Nota Estudiante	155
Figura B. 31 Paquete Modelo Psicología DOBE.....	155
Figura B. 32 Caso de Uso Gestionar Ficha Psicológica	157
Figura B. 33 Caso de Uso Gestionar Terapia	160

Figura B. 34 Caso de Uso Gestionar Motivo.....	163
Figura B. 35 Caso de Uso Gestionar Visita	165
Figura B. 36 Caso de Uso Gestionar Caso.....	167
Figura E. 1 Clases de prueba Modulo Administración	170
Figura E. 2 Prueba Método Asignar Perfil Módulo Adicional	170
Figura E. 3 Ejecución prueba Asignar Perfil Módulo Adicional	170
Figura E. 4 Resultado prueba Asignar Perfil Módulo Adicional.....	170
Figura E. 5 Prueba Método Opción Usuario	171
Figura E. 6 Ejecución prueba Opción Usuario	171
Figura E. 7 Resultado prueba Opción Usuario	171
Figura E. 8 Resultado prueba Obtener Perfil Opción.....	171
Figura E. 9 Resultado prueba Gestión Opción Perfil	171
Figura E. 10 Resultado prueba Método Gestión Perfil.....	172
Figura E. 11 Resultado prueba Obtener Detalle Perfil.....	172
Figura E. 12. Resultado prueba Buscar Usuario.....	172
Figura E. 13 Resultado prueba Buscar Usuario por Perfil	172
Figura E. 14 Resultado prueba Cambiar Password	172
Figura E. 15 Resultado prueba Cifrar Clave	172
Figura E. 16 Resultado prueba Quitar Acceso.....	173
Figura E. 17 Prueba Método Gestión Usuario	173
Figura E. 18 Resultado prueba Gestión Usuario.....	173
Figura E. 19 Resultado prueba Grabar Log Usuario.....	173
Figura E. 20 Resultado prueba Obtener Parámetros Seguridad.....	173
Figura E. 21 Resultado prueba Obtener Perfil	174
Figura E. 22 Resultado prueba Resetear Password	174
Figura E. 23 Resultado prueba Gestión Barrio	174
Figura E. 24 Resultado prueba Gestión Cantón	174
Figura E. 25 Prueba Método Buscar Estudiante	175
Figura E. 26 Resultado prueba Buscar Estudiante	175
Figura E. 27 Resultado prueba Familia por estudiante	175
Figura E. 28 Resultado prueba Gestión Estudiante por Familia	175
Figura E. 29 Resultado prueba Gestión País.....	176
Figura E. 30 Resultado prueba Gestión Parentesco.....	176
Figura E. 31 Resultado prueba Gestión Parroquia	176
Figura E. 32 Resultado prueba Buscar Personal.....	176
Figura E. 33 Prueba Método Gestión Personal.....	177
Figura E. 34 Resultado prueba Gestión Personal.....	177
Figura E. 35 Resultado prueba Gestión Personal Capacitación	177
Figura E. 36 Resultado prueba Gestión Provincia	177
Figura E. 37 Resultado prueba Gestión Raza	178
Figura E. 38 Resultado prueba Gestión Tipo Domicilio	178
Figura E. 39 Resultado prueba Gestión Tipo Personal.....	178

Figura E. 40 Resultado prueba Gestión Teléfono	178
Figura E. 41 Resultado prueba Gestión Ubicación	178
Figura E. 42 Resultado prueba Gestión Área Académica.....	179
Figura E. 43 Resultado prueba Año Lectivo Curso	179
Figura E. 44 Resultado prueba Año Lectivo Paralelo Curso	179
Figura E. 45 Resultado prueba Gestión Curso Paralelo	179
Figura E. 46 Resultado prueba Gestión Curso	179
Figura E. 47 Resultado prueba Obtener Curso.....	180
Figura E. 48 Resultado prueba Obtener Curso Materia	180
Figura E. 49 Resultado prueba Gestión Especialidad.....	180
Figura E. 50 Ejecución prueba Listado Especialidad	180
Figura E. 51 Resultado prueba Listado Especialidad	180
Figura E. 52 Resultado prueba Estructura Año Actual.....	181
Figura E. 53 Resultado prueba Estructura Año Lectivo	181
Figura E. 54 Resultado prueba Parcial Año Lectivo.....	181
Figura E. 55 Resultado prueba Valor Académico Año Lectivo	181
Figura E. 56 Valor Conducta Año Lectivo	181
Figura E. 57 Resultado prueba Fecha Año Lectivo Vigencia	181
Figura E. 58 Resultado prueba Estructura Año Unifica.....	182
Figura E. 59 Resultado prueba Estructura Curso Paralelo	182
Figura E. 60 Resultado prueba Estructura Nivel.....	182
Figura E. 61 Resultado prueba Fuera Horario Clase	182
Figura E. 62 Prueba Método Gestión Horario Clase.....	183
Figura E. 63 Resultado prueba Gestión Horario Clase	183
Figura E. 64 Resultado prueba Horario Clase Docente Materia	183
Figura E. 65 Resultado prueba Obtener Horario Clase Curso Paralelo.....	183
Figura E. 66 Resultado prueba Obtener Materia Docente Curso	184
Figura E. 67 Resultado prueba Verifica Configuración Hora Día	184
Figura E. 68 Ejecución prueba Gestionar Materia Curso	184
Figura E. 69 Resultado prueba Gestión Materia	184
Figura E. 70 Resultado prueba Gestión Nivel	184
Figura E. 71 Ejecución prueba Estudiante Nota	185
Figura E. 72 Resultado prueba Gestión Nota Estudiante Materia.....	185
Figura E. 73 Resultado prueba Elimina Estudiante Orienta Tarea.....	185
Figura E. 74 Ejecución prueba Elimina Estudiante Vive con	185
Figura E. 75 Ejecución prueba Gestión Estudiante Vive Con	185
Figura E. 76 Resultado prueba Gestión Ficha Estudiante	186
Figura E. 77 Resultado prueba Gestión Orienta Tarea.....	186
Figura E. 78 Resultado prueba Obtener Ficha Estudiante.....	186

LISTADO DE ANEXOS

Anexo A Especificación de Requerimientos según Norma IEEE 830	69
Anexo B Especificación de Casos de Uso y Diagramas	93
Anexo C Diagrama Conceptual de la Base de Datos	167
Anexo D Diagrama Físico de la Base de Datos	168
Anexo E Pruebas Unitarias	169

RESUMEN

En el “Colegio Militar Abdón Calderón No 10” se utilizaba una aplicación Windows Cliente – Servidor antigua que generaba muchos problemas de rendimiento, inconsistencia y confiabilidad de la información, debido a la instalación de varios parches y a la utilización de tecnología antigua. Además, varios procesos eran realizados de manera manual, complicando significativamente los procesos administrativos de la institución. Se realizó el Análisis, Diseño, Desarrollo, pruebas e implementación de los Módulos Orientados a la WEB, dicho objetivo fue alcanzado utilizando la metodología de desarrollo de aplicaciones AUP (Proceso Unificado Ágil), el cual permite un desarrollo simple y fácil de entender las aplicaciones de software. Además, con la ayuda de herramientas CASE se logró culminar con todas las etapas de la metodología. Una vez en producción los procesos se realizan de manera eficiente, la integridad de los datos está asegurada. Además, con una alta disponibilidad, debido a que la aplicación se encuentra en una IP Pública, esta puede ser accedida desde cualquier PC que disponga de internet. Así, acercamos la aplicación a todas las partes involucradas como son: docentes, psicólogos, personal administrativo, estudiantes y padres de familia, a que sean parte activa de la correcta utilización de los módulos desarrollados.

PALABRAS CLAVES:

- **WINDOWS**
- **APLICACIÓN WEB**
- **PARTES INVOLUCRADAS**
- **COLEGIO**
- **AUP**

ABSTRACT

In the "Military College Abdon Calderon No 10 " An old client server application is used, that generated many performance problems, inconsistencies and reliability of the information, due to the installation of several patches and the use of old technology. In addition, several processes were performed manually, significantly complicating the administrative processes of the institution. Analysis, design, development, testing and implementation of WEB-oriented modules to be made, this goal was achieved using the methodology of application development AUP (Agile Unified Process), which allows a simple and easy to understand development of software applications. In addition, with the help of CASE tools it was achieved culminate in all stages of the methodology. Once in production processes are carried out more efficiently, the data integrity is assured. Furthermore, high availability, because the application is in a public IP, This can be accessed from any PC with Internet. So, we approach the application to stakeholders such as: teachers, psychologists, administrators, students and parents, to make them part of the proper use active developed modules.

KEYWORDS:

- **WINDOWS**
- **APPLICATION WEB**
- **STAKEHOLDERS**
- **FACULTY**
- **AUP**

CAPÍTULO 1

1.1. Introducción

La historia de la Institución comienza con la creación de la Casa Maternal Militar N° 1, el 28 de Noviembre de 1.953. Por disposición de la Dirección de Educación de la Fuerza Terrestre el Colegio "Abdón Calderón", la Escuela y el Jardín de Infantes se unifican apareciendo legalmente como Unidad Educativa "Abdón Calderón". En Febrero de 1.999 el Comando General de la Fuerza Terrestre decide convertir a esta Unidad Educativa en Colegio Militar.

La Institución cuenta con dos locales los mismos que funcionan de la siguiente manera: la 1ra instalación de 1ro a 7mo de Educación General Básica (EGB) Jardín y escuela ubicada en la avenida Maldonado Sector la Recoleta. De 8vo a 3ro de bachillerato funciona en la avenida Mariscal Sucre y Michelena Sector Pintado.

El Colegio Militar No 10 Abdón Calderón impactado por el desarrollo vertiginoso de las nuevas tecnologías, está en la obligación de mantenerse a la vanguardia.

Por ello propone desarrollar tres módulos (Administrativo, Académico, DOBE) que permita a los inspectores, psicólogos, trabajadoras sociales, docentes, personal administrativo, cadetes y representantes contar con procesos automatizados que ayuden a una mejor organización de los departamentos.

1.2. Planteamiento del problema

La aplicación con la que actualmente trabaja la Institución constas de tres módulos (Secretaría, Académico, Colectaría) es una aplicación Windows Cliente - Servidor, desarrollada en Delphi 6.0 en el año 1999 - 2000 con SQL Anywhere 6.0 con motor de base de datos.

La aplicación se encuentra instalada en dos ubicaciones, que funcionan independientemente una de la otra.

Con el transcurso del tiempo se ha hecho evidente problemas como:

- Reducción del rendimiento debido al crecimiento de la base de datos.
- Instalación de parches debido a los constantes cambios institucionales y gubernamentales.
- Inconsistencia y perdida de la información: debido al traspaso de la misma en medios de almacenamiento externos y llevando a la matriz, donde luego es cargada en la base central, con lo cual la consistencia de la información se ve comprometida.
- Ineficiencia: los procesos son lentos y no son en tiempo real, hasta que la información sea cargada en la base central.
- No se dispone de un módulo de Administración que permita la gestión de usuarios, contraseñas, perfiles de acceso.

Aparte de lo expuesto anteriormente, existen muchas actividades manuales o semiautomáticas que realiza la Inspección General, personal administrativo y Departamento de Orientación y Bienestar Estudiantil (DOBE), algunas de estos problemas son:

DOBE

- La ficha psicológica de los cadetes es llevada en una tabla de Microsoft Access 2003.
- Seguimiento de casos especiales de los cadetes.
- Registro de terapias de los cadetes.
- Registro de visitas domiciliarias a los padres de familia.

- Los informes de casos atendidos a los cadetes es realizada en hojas de Microsoft Excel.
- No existe estadísticas de casos atendidos a los cadetes.

ACADÉMICO:

- La estructura del año lectivo (aportes) no se acopla a la estructura vigente ordenada por el ministerio de Educación.
- No existe el subproceso de horario de clases que servirá como insumo para el módulo de inspección.
- Los representantes no pueden verificar el progreso de los estudiantes.
- La ficha psicológica no está integrada.
- Debe mejorarse el subproceso de ubicabilidad de los cadetes, personal.
- Cualquier usuario puede cambiar la información.
- El sistema actual no contempla reportes para la toma de decisiones.

1.3. Justificación

Por lo expuesto anteriormente el presente Plan propone desarrollar tres módulos (Administración, Académico, DOBE) que permita a los inspectores, psicólogos, trabajadoras sociales, docentes, personal administrativo, cadetes y representantes contar con procesos automatizados que ayuden a una mejor organización de los departamentos, lo cual contribuirá a la toma de decisiones de manera oportuna en beneficio de todos involucrados con la institución.

1.4. Objetivos

1.4.1 Objetivo General

Realizar el análisis, diseño, implementación de los módulos de Administración, Académico y DOBE orientados a la WEB para el Colegio Militar Abdón Calderón N° 10.

1.4.2 Objetivos Específicos

- Realizar la especificación de requerimientos según el estándar de IEEE 830 para el desarrollo de los módulos.
- Realizar Análisis, Diseño, Desarrollo de los módulos de Administración, Académico y DOBE, según la metodología seleccionada.
- Probar e implementar los módulos a desarrollar, con todas las funcionalidades que requiere la institución.

1.5 Alcance

La Solución propuesta para el colegio Militar Abdón Calderón N° 10. es el desarrollo e implementación de tres módulos Administración, Académico y DOBE los cuales serán desarrollados en un ambiente Web que trabajará a nivel de intranet e internet para la sección de Educación General Básica y Bachillerato.

1.6 Beneficios para el Comil 10

- Integridad de los datos.
- Confidencialidad de los datos.
- Alta disponibilidad de la información, al utilizar un motor de base de datos de última generación.
- Utilizar tecnologías de punta.

- Rapidez de la información.
- Generar reportes con tecnología de punta.
- Brindar un servicio de calidad.
- Ofrecer a los usuarios un software de calidad.
- Sentar la base para la integración completa de todos los módulos utilizados por la Institución.
- Utilizar herramientas libres para el desarrollo de los módulos.
- Fomentar el uso de las tecnologías de la información en la gestión de los procesos educativos.
- Mejorar la imagen institucional del Colegio Abdón Calderón.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Introducción a AUP (Proceso Unificado Ágil) (Ambler, 2006) (Flores Ervin, 2013)

AUP nace en 2005 en manos del desarrollador Scott W. Ambler el cual lo define de la siguiente manera: El proceso unificado ágil (AUP) es un proceso de desarrollo de programas basado en el Proceso Unificado Rational de IBM (Rational Unified Process-RUP). El ciclo de vida de AUP es secuencial en lo grande e iterativo en lo pequeño, entregando productos incrementales en un cierto plazo.

2.1.1. AUP

Describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP¹. El AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (Test Driven Development - TDD), Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad.

AUP² se preocupa especialmente de la gestión de riesgos. Propone que aquellos elementos con alto riesgo obtengan prioridad en el proceso de desarrollo y sean abordados en etapas tempranas del mismo. Para ello, se

¹ Proceso Unificado de Rational

² Proceso Unificado Ágil

crean y mantienen listas identificando los riesgos desde etapas iniciales del proyecto.

Especialmente relevante en este sentido es el desarrollo de prototipos ejecutables durante la fase de elaboración del producto, donde se demuestre la validez de la arquitectura para los requisitos clave del producto y que determinan los riesgos técnicos.

El proceso AUP establece un Modelo más simple que el que aparece en RUP, por lo que reúne en una única disciplina de Modelado de Negocio los flujos de Requisitos y Análisis y Diseño. El resto de disciplinas (Implementación, Pruebas, Despliegue, Gestión de Configuración, Gestión de Proyectos y Ambiente) coinciden con las restantes de RUP.

2.1.2. Ciclo de vida del Proceso Unificado Ágil (AUP)

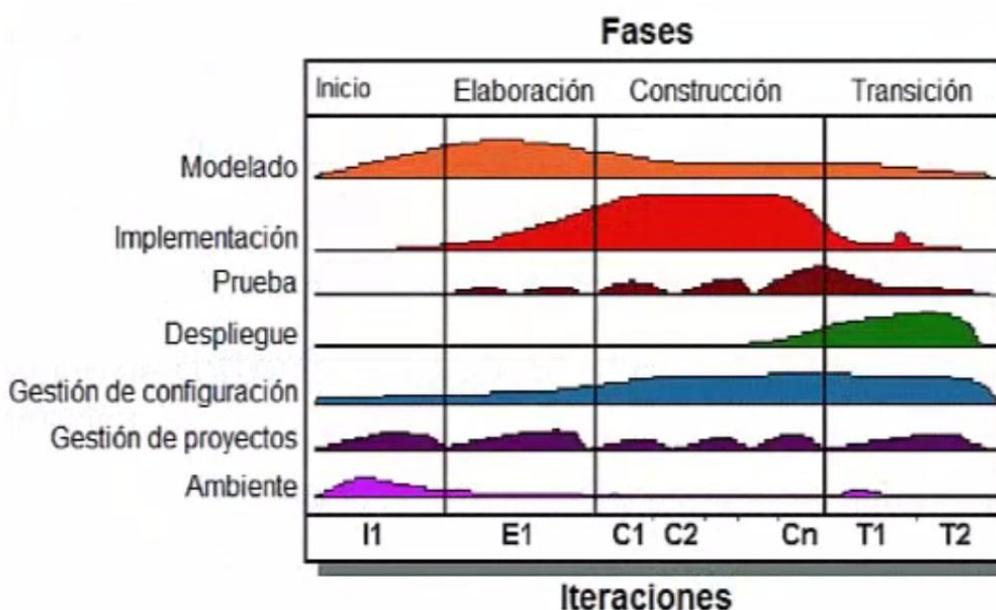


Figura 2. 1 AUP (Fases de Ciclo de vida)

2.1.3. Principios de AUP

AUP se basa en los siguientes principios:

- Su personal sabe lo que está haciendo: La gente no va a leer la documentación de proceso detallada, sino que ella querrá una

cierta dirección de alto nivel y/o el entrenamiento de vez en cuando.

- Simplicidad: Se describe usando poca cantidad de páginas.
- Agilidad: Se ajusta a los valores y a los principios de la Agile Alliance.
- Independencia de la herramienta: Se puede utilizar cualquier juego de herramientas, pero se recomiendan herramientas simples.

2.1.4. Fases

AUP cuenta con las siguientes fases:

Inicio

Objetivo: Identificar el alcance inicial del proyecto, proveer una arquitectura potencial para el sistema, y obtener un financiamiento inicial del proyecto y la aceptación de los stakeholders.

Tareas:

- Definir alcance del proyecto
- Estimar costos y plazos
- Definir riesgos
- Determinar factibilidad del proyecto
- Preparar el ambiente

Hito: Objetivos del ciclo de vida (Life Cycle Objectives - LCO).

Elaboración

Objetivo: obtener la arquitectura del sistema.

Tareas:

- Identificar arquitectura
- Validar la arquitectura
- Desarrollar el ambiente el proyecto
- Seleccionar el personal del proyecto

Hito: Arquitectura del ciclo de vida (LCA).

Construcción

Objetivo: implementar un software sobre una base incremental la que debe estar relacionada con los objetivos de los involucrados.

Tarea:

- Modelado, construcción y testeo del sistema
- Creación de la documentación de apoyo

Hito: Capacidad operacional inicial (IOC).

Transición

Objetivo: validar y entregar el sistema en un ambiente de producción.

Tarea:

- Test del sistema
- Test de usuarios
- Retrabajo del sistema
- Instalación del sistema

Hito: Lanzamiento del producto (PR).

2.1.5. Disciplinas

Definen actividades que el equipo de desarrolladores debe realizar para construir, validar y entregar un software que satisfaga las necesidades de los stakeholders.

Modelo

Tiene como fines entender los procesos de negocios de la organización, el dominio de problema que puede ser abordado por el software e identificar una solución viable.

Implementación

Transformar los modelos en código ejecutable y aplicar pruebas básicas en unidades particulares de prueba.

Prueba

Realizar una evaluación objetiva para asegurar la calidad. Esto incluye encontrar defectos, validar que el sistema funcione como fue diseñado, y verificar que los requerimientos estén abordados por las funcionalidades.

Despliegue

Planificar la entrega del sistema y ejecutar el plan para que el sistema esté disponible para los usuarios.

Administración de la Configuración

Administrar el acceso a los artefactos del proyecto. Esto no solo incluye el seguimiento de las versiones de los artefactos, sino también controlar y administrar los cambios sobre ellos.

Administración del Proyecto

Dirigir las actividades que forman parte del proyecto. Esto incluye administración de riesgos, dirigir personas y coordinar personas con sistemas que están fuera del alcance del proyecto.

Ambiente

Facilitar todo el entorno que permita el normal desarrollo del proyecto.

2.1.6. Iteraciones

Los equipos del AUP entregan típicamente lanzamientos del desarrollo en el final de cada iteración. Como se puede observar en la Figura 2.2 el primer release puede tomar más tiempo que el segundo; eso se debe en que en primera instancia hay más imprevistos a resolver que en las instancias futuras. Como se puede observar, lo que propone AUP es que a medida que el proyecto avanza las iteraciones se vuelvan más cortas.

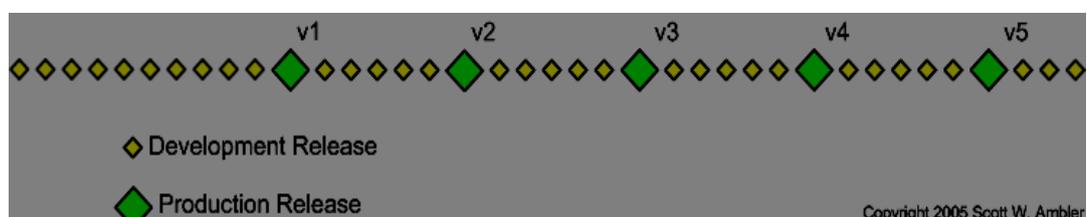


Figura 2. 2 Iteraciones para los Releases de Producción

2.1.7. Roles de equipo

Los principios sobre los que se basan AUP son:

- a) Los roles pueden llevarse a cabo por varias personas.
- b) Una persona puede adquirir roles múltiples.
- c) Un rol no es una posición.

Entre los roles más importantes se tienen:

Administrador de la Base de Datos Ágil: Administrador de la base de datos que trabaja con el equipo para diseñar, probar, evolucionar y soportar el esquema de datos de la aplicación

Modelador Ágil: Crea y evoluciona los modelos, de forma colaborativa con los otros miembros del equipo, haciendo uso de herramientas CASE y de otras herramientas. Los modelos ágiles son simplemente escasos, buenos y suficientes.

Administrador de la Configuración: Es responsable de proveer la infraestructura y ambiente de Gestión de Configuración para el desarrollo del equipo.

Instalador: Es responsable para desplegar el sistema en el ambiente de pre-producción y producción.

Desarrollador: Escribe, prueba y construye software.

Ingeniero del proceso: Desarrolla, personaliza y soporta los materiales del proceso de software de la organización.

Administrador del proyecto: Administra el equipo del proyecto, defiende a los miembros del equipo, construye las relaciones con los involucrados, coordina la interacción con ellos, planifica, administra y asigna recursos, establece prioridades y mantiene al equipo enfocado

Revisor: Evalúa los productos de trabajo del proyecto, frecuentemente trabaja con el progreso del proyecto y retroalimenta al equipo

Involucrado: Cualquiera que sea usuario directo o indirecto, administrador de usuario, gerente de más nivel, miembro del equipo

de operaciones, miembro del equipo de soporte (help desk), desarrolladores de otros sistemas que integran o interactúan con el que se desarrolla, personal de mantenimiento potencialmente afectado por el desarrollo o instalación del sistema bajo desarrollo.

Documentador técnico: Responsable de obtener la documentación de usuario (materiales de entrenamiento, documentación de operación, documentación de soporte y documentación de usuario)

Administrador de pruebas: Es responsable por el éxito del esfuerzo de las pruebas e incluye planificación, administración, y defensa de la calidad y de las pruebas

Probador: Responsable por la escritura, conducción y anotación de los resultados de las pruebas

Especialista de herramientas: Es responsable de la selección, adquisición, configuración y soporte de las herramientas.

2.1.8. Entregables

AUP clasifica los entregables en tres tipos:

Productos a entregar mínimos: es la documentación mínima requerida para realizar el proyecto.

- Otros productos del trabajo de proyecto: es la documentación no esencial según AUP.
- Productos del trabajo de la empresa: son los entregables que la empresa debe realizar para el proyecto.

Los productos a entregar mínimos son los siguientes:

Sistema: El software, el soporte físico, y la documentación de trabajo que se desplegará en la producción.

Código fuente: El código del programa para su sistema.

Casos de testeo: Una colección de casos de prueba, y el código para hacerlos funcionar en el orden apropiado.

Scripts de Instalación: Código para instalar el sistema en su ambiente de producción.

Documentación de sistema: La documentación entregada como parte de su sistema para ayudar a los stakeholders a trabajar con él y los developers para mantenerlo y desarrollarlo.

Release Notes: Resumen las “buenas cosas para saber” sobre la versión actual del sistema que está construyendo.

Modelo de Requerimientos: Test de Aceptación, Procesos de Negocio, Dominio, Casos de Uso, Interfaz de Usuario. Modelo del diseño describe el diseño del sistema.

El mejor lugar para documentar el diseño es en los test unitarios y en el código fuente

Ventajas

- El personal sabe lo que está haciendo: no obliga a conocer detalles.
- Simplicidad: apuntes concisos.
- Agilidad: procesos simplificados del RUP
- Centrarse en actividades de alto valor: esenciales para el desarrollo.
- Herramientas independientes: a disposición del usuario.
- Fácil adaptación de este producto: de fácil acomodo (HTML)

Desventajas

- El AUP es un producto muy pesado en relación al RUP.
- Como es un proceso simplificado, muchos desarrolladores eligen trabajar con el RUP, por tener a disposición más detalles en el proceso.

2.2 Lenguaje Unificado de Modelado (UML)

2.2.1. Definición de UML

UML es ante todo un lenguaje. Un lenguaje proporciona un vocabulario y unas reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema, e indica cómo crear y leer los modelos, pero no dice cómo crearlos.

Los objetivos de UML son muchos, pero se pueden sintetizar en funciones:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Aunque UML está pensado para modelar sistemas complejos con gran cantidad de software, el lenguaje es lo suficientemente expresivo como para modelar sistemas que no son informáticos, como flujos de trabajo (*workflow*) en una empresa, diseño de la estructura de una organización y por supuesto, en el diseño de hardware.

Un modelo UML está compuesto por tres clases de bloques de construcción:

- Elementos: Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.)
- Relaciones: relacionan los elementos entre sí.
- Diagramas: Son colecciones de elementos con sus relaciones.

2.2.2 Diagramas UML

Un diagrama es la representación gráfica de un conjunto de elementos con sus relaciones. En concreto, un diagrama ofrece una vista del sistema a modelar.

Para poder representar correctamente un sistema, UML ofrece una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas. UML incluye los siguientes diagramas:

- Diagrama de Clases: Muestra una colección de elementos de modelado declarativo (estáticos), tales como clases, tipos y sus contenidos y relaciones.
- Diagrama de Componentes: Representa los componentes que componen una aplicación, sistema o empresa. Los componentes contienen relaciones, interacciones y sus interfaces públicas.
- Diagrama de Estructura de Composición: Representa la estructura interna de un clasificador (tal como una clase, un componente o un caso de uso), incluyendo los puntos de interacción del clasificador con otras partes del sistema.
- Diagrama de Despliegue Físico: Un diagrama de despliegue físico muestra cómo y dónde se desplegará el sistema. Las máquinas físicas y los procesadores se representan como nodos y la construcción interna puede ser representada por nodos o artefactos embebidos. Como los artefactos se ubican en los nodos para modelar el despliegue del sistema, la ubicación es guiada por el uso de las especificaciones de despliegue.
- Diagrama de Objetos: Diagrama que presenta los objetos y sus relaciones en un punto del tiempo. Un diagrama de objetos se puede considerar como un caso especial de un diagrama de clases o un diagrama de comunicaciones.
- Diagrama de Paquetes: Un diagrama que presenta cómo se organizan los elementos de modelado en paquetes y las dependencias entre ellos, incluyendo importaciones y extensiones de paquetes.
- Diagrama de Actividades: Representa los procesos de negocios de alto nivel, incluidos el flujo de datos. También

puede utilizarse para modelar lógica compleja y/o paralela dentro de un sistema.

- Diagrama de Comunicaciones: (anteriormente: Diagrama de Colaboraciones). Es un diagrama que enfoca la interacción entre líneas de vida, donde es central la arquitectura de la estructura interna y cómo ella se corresponde con el paso de mensajes. La secuencia de los mensajes se da a través de un esquema de numerado de la secuencia.
- Diagrama de Revisión de la Interacción: Los Diagramas de Revisión de la Interacción enfocan la revisión del flujo de control, donde los nodos son Interacciones u Ocurrencias de Interacciones. Las Líneas de Vida de los Mensajes no aparecen en este nivel de revisión
- Diagrama de Secuencias: Un diagrama que representa una interacción, poniendo el foco en la secuencia de los mensajes que se intercambian, junto con sus correspondientes ocurrencias de eventos en las Líneas de Vida.
- Diagrama de Máquinas de Estado: Un diagrama de Máquina de Estados ilustra cómo un elemento, muchas veces una clase, se puede mover entre estados que clasifican su comportamiento, de acuerdo con disparadores de transiciones, guardias de restricciones y otros aspectos de los diagramas de Máquinas de Estados, que representan y explican el movimiento y el comportamiento.
- Diagrama de Tiempos: El propósito primario del diagrama de tiempos es mostrar los cambios en el estado o la condición de una línea de vida (representando una Instancia de un Clasificador o un Rol de un clasificador) a lo largo del tiempo lineal. El uso más común es mostrar el cambio de estado de un objeto a lo largo del tiempo, en respuesta a los eventos o estímulos aceptados. Los eventos que se reciben se anotan a

medida que muestran cuándo se desea mostrar el evento que causa el cambio en la condición o en el estado.

- Diagrama de Casos de Uso: Un diagrama que muestra las relaciones entre los actores y el sujeto (sistema), y los casos de uso.

(Aplicaciones en Capas, 2013) (Manual de Umbrello UML Modeller, 2013) (Manual de Umbrello UML Modeller, 2013).

2.3. Modelo tres capas de una Aplicación Web

La arquitectura basada en capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo una forma muy efectiva de separación de responsabilidades. El rol indica el modo y tipo de interacción con otras capas, y la responsabilidad indica la funcionalidad que está siendo desarrollada.

Por ejemplo, una aplicación web típica está compuesta por una capa de presentación (funcionalidad relacionada con la interfaz de usuario), una capa de negocios (procesamiento de reglas de negocios) y una capa de datos (funcionalidad relacionada con el acceso a datos), tal como se muestra en la Figura 2.3.

El estilo de arquitectura basado en capas se identifica por las siguientes características:

- Describe la descomposición de servicios de forma que la mayoría de la interacción ocurre solamente entre capas vecinas.
- Las capas de una aplicación pueden residir en la misma máquina física (misma capa) o puede estar distribuido sobre diferentes computadores (n-capas).
- Los componentes de cada capa se comunican con otros componentes en otras capas a través de interfaces muy bien definidas.

- Este modelo ha sido descrito como una “pirámide invertida de re-uso” donde cada capa agrega responsabilidad y abstracción a la capa directamente sobre ella.

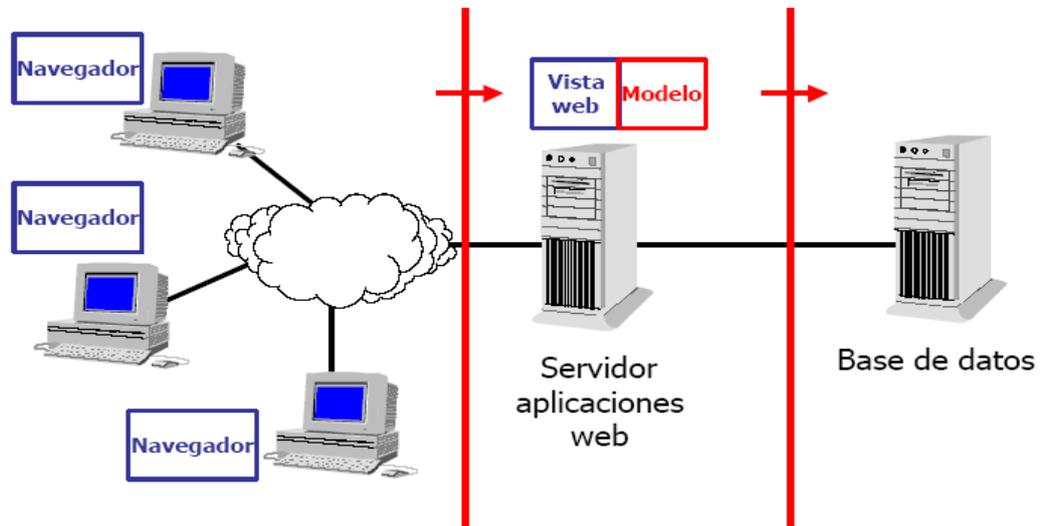


Figura 2. 3 Arquitectura Web en Tres Capas

2.4. Seguridad en Aplicaciones Web

Las aplicaciones Web deben cumplir con las siguientes características de seguridad:

- Disponibilidad: Propiedad o característica de los activos, consistente en que las entidades o procesos autorizados tienen acceso a los mismos cuando lo requieren.
- Autenticidad: Propiedad o característica consistente en que una entidad es quien dice ser o bien que garantiza la fuente de la que proceden los datos
- Integridad: Propiedad o característica consistente en que el activo de información no ha sido alterado de manera no autorizada.
- Confidencialidad: Propiedad o característica consistente en que la información ni se pone a disposición, ni se revela a individuos, entidades o procesos no autorizados.
- Trazabilidad: Propiedad o característica consistente en que las actuaciones de una entidad pueden ser imputadas exclusivamente a dicha entidad.

2.4.1 Medidas de prevención contra las vulnerabilidades

El análisis de las vulnerabilidades potenciales es un objetivo básico para el incremento de la seguridad en las aplicaciones web, que en ocasiones es subestimado como factor de riesgo crítico. El mantener parámetros que no son verificados, roles sin controlar, desbordamientos que se producen en la memoria son algunas de las situaciones que pueden provocar brechas de seguridad en las aplicaciones.

Los desarrollos que se realizan comercialmente presentan las mismas deficiencias. De ahí, que necesiten una actualización constante para asegurar la reparación de bugs que se van encontrando en el tiempo de vida de las aplicaciones. Con frecuencia, se asemeja la seguridad de una aplicación con la seguridad de la plataforma donde se ejecuta. El esfuerzo que se realice para aumentar la seguridad tanto a nivel de desarrollo como de diseño, debe de ser un esfuerzo a nivel de grupo. Hay que incluir en el desarrollo un compromiso para el aseguramiento de las aplicaciones. Los servidores de producción y los otros sistemas deben mantenerse regularmente con las últimas firmas para garantizar que están libres de vulnerabilidades a nivel de sistema. Se recomienda seguir las siguientes indicaciones destinadas a mantener una aplicación alejada de las vulnerabilidades:

- **Entornos de trabajo diferenciados.** Es muy aconsejable mantener entornos separados del de producción. Los entornos de calidad y de desarrollo son frecuentemente manipulados por lo que entrarán en conflicto con los datos mostrados por producción. Es primordial que sus entornos estén bien diferenciados, a ser posible mediante un cortafuego. Esta situación es prioritaria cuando el entorno de producción es una zona interna o de transición y accesible desde Internet. Si el entorno de producción tramita y ejecuta procesos, hay que asegurar los mismos, tanto a nivel de sistema como de aplicación antes de incluir nuevos procesos. Este principio hay que mantenerlo cada vez que se actualice o mejore.

- **Distribución de las actualizaciones.** Con cierta frecuencia las aplicaciones realizan actualizaciones para cubrir posibles deficiencias que se detectan a lo largo del ciclo de vida de la aplicación. Dependiendo del carácter de estas aplicaciones, si son externas, las actualizaciones son inmediatas, si son internas se debe además examinar el código de las aplicaciones creadas internamente e introducir las actualizaciones o nuevas versiones a medida que vaya siendo necesario.
- **Contra medidas temporales.** Es necesario realizar un protocolo de respuesta ante posibles situaciones que entrañen riesgo para las aplicaciones. En este protocolo se pueden indicar acciones a realizar en función de la gravedad del riesgo detectado (cerrar puertos, bloquear direcciones, routers, etc.). Es muy importante, para lograr eficiencia en la respuesta, saber interpretar los efectos que delimitan el riesgo y mantener a todo el equipo de trabajo involucrado plenamente e informado del protocolo de respuesta ante la activación del mismo.
- **Modo de Fallos.** Se debe disponer de un plan que permita mantener en funcionamiento el proceso crítico, si se da la situación de que una aplicación falle. Es decir, debe de permitir una transacción que permita enrutarse, de forma ajena a la aplicación, o bien permitir realizar de forma manual las funciones descritas en la aplicación. Lo más importante es mantener el criterio de " cierre ante fallo". Una aplicación no debe abrirse si detecta un fallo. Si lo hace, corre el riesgo de permitir todo tipo de acciones en el sistema. Las aplicaciones o sus comprobaciones internas siempre deben cerrarse al fallar.
- **Conocimiento de desarrolladores y políticas.** Es muy importante que todas las personas involucradas en un desarrollo tengan el conocimiento necesario de las herramientas disponibles y de las políticas necesarias para crear códigos seguros. Las políticas de

seguridad deben de ser generales, por lo tanto deben de mantenerse en un nivel que permita aplicarse al global del conjunto de proyectos.

2.5. Ventajas de una Aplicación Web

- **Compatibilidad multiplataforma:** las diferentes tecnologías utilizadas para el desarrollo de aplicaciones web como PHP, Java, AJAX, JavaScript, MySQL, PostgreSQL, SQLServer y Oracle permiten un desarrollo efectivo y robusto de las mismas, por lo cual pueden ser ejecutadas en los principales sistemas operativos sin ninguna restricción, a diferencia del software de escritorio en donde debe existir una versión instalable diferente para cada uno de ellos.
- **Menos requerimientos de hardware:** las aplicaciones web consumen menos recursos hardware de un computador que los programas instalados localmente, en cuanto a memoria RAM y espacio en disco duro. Esto se debe a que las aplicaciones se alojan y ejecutan en los servidores del proveedor o de la empresa, dejando más espacio para correr múltiples aplicaciones al mismo tiempo, sin deteriorar el rendimiento de los equipos del usuario final.
- **Acceso inmediato:** dado que las aplicaciones web no necesitan ser descargadas, instaladas y configuradas en cada uno de los equipos, un usuario puede acceder de forma remota mediante un navegador Web y comenzar a trabajar sin importar la configuración y el hardware de su equipo.
- **Múltiples usuarios concurrentes:** las aplicaciones web pueden ser utilizadas por múltiples usuarios al mismo tiempo, con lo cual diferentes usuarios pueden ver e incluso editar toda la información de manera conjunta.
- **Información en línea:** las aplicaciones web permiten acceder a la información almacenada en los servidores desde un computador conectado a Internet o a su red local de una forma segura, por lo cual

puede disponer de la misma en sus desplazamientos y desde cualquier lugar del mundo, dado que cuenta con una información centralizada almacenada en bases de datos remotas.

- Información más segura: dado que la información se almacena de forma centralizada y a las copias de seguridad de los servidores, los usuarios van a tener mucho menos riesgo de perder sus datos ocasionado por un daño en sus discos duros o un virus en su computador.
- Menos errores y fallas (bugs): las aplicaciones web son menos propensas a crear problemas técnicos debido a conflictos con el software, hardware o con otras aplicaciones existentes. Con las aplicaciones web todos los usuarios utilizan la misma versión y por ende los posibles bugs pueden ser corregidos tan pronto como sean descubiertos, y ningún usuario habrá actualizado individualmente la aplicación.
- Facilidad de actualización: cuando una aplicación web debe ser actualizada este procedimiento (Internet Ya, 2013) se realiza una vez para todos los usuarios, por lo cual no es necesario que cada usuario tome acciones pro-activas que interfieran con sus hábitos de trabajo al iniciar nuevas descargas y procedimientos de instalación individuales.

2.6. Desventajas de una Aplicación Web (Alegsa, 2013)

- Las aplicaciones web requieren navegadores web totalmente compatibles para funcionar. Incluso muchas veces requieren las extensiones apropiadas y actualizadas para operar.
- Muchas veces requieren una conexión a internet para funcionar, si la misma se interrumpe, no es posible utilizarla más. De todas maneras, en ocasiones, pueden ser descargadas e instaladas localmente para su uso offline.
- Muchas no son de código abierto, perdiendo flexibilidad.

- La aplicación web desaparece si así lo requiere el desarrollador o si el mismo se extingue. Las aplicaciones tradicionales, en general, pueden seguir usándose en esos casos.
- El usuario, en general, no tiene libertad de elegir la versión de la aplicación web que quiere usar. Un usuario podría preferir usar una versión más antigua, hasta que la nueva sea probada.
- En teoría, el desarrollador de la aplicación web puede rastrear cualquier actividad que el usuario haga. Esto puede traer problemas de privacidad.

2.7. Herramientas, lenguajes de programación y servicios

2.7.1. Microsoft Visual Studio Express Edition

Es un programa de desarrollo en entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows desarrollado y distribuido por Microsoft Corporation. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic.NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros. Es de carácter gratuito.

2.7.2. Microsoft SQL Server 2008 Express

Es una plataforma de base de datos que se basa en Microsoft SQL Server. SQL Server Express facilita el desarrollo de aplicaciones controladas por datos con gran variedad de funciones, que mejoran la seguridad del almacenamiento y se implementan con rapidez.

2.7.3. Internet Information Services o IIS

Es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Originalmente era parte del Option Pack para Windows NT. Este servicio convierte a una PC en un servidor web para Internet o una intranet, es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente.

2.7.4. JavaScript

Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente.

2.7.5. Lenguaje C#

Es un lenguaje de programación que se ha diseñado para generar diversas aplicaciones que se ejecutan en .NET Framework. C# es simple, eficaz, con seguridad de tipos y orientado a objetos. Las numerosas innovaciones de C# permiten desarrollar aplicaciones rápidamente y mantener la expresividad y elegancia de los lenguajes de estilo de C.

2.7.6. Transact-SQL (T-SQL)

Es una extensión al SQL de Microsoft y Sybase. SQL, que frecuentemente se dice ser un Lenguaje de Búsquedas Estructurado (por sus siglas en inglés), es un lenguaje de cómputo estandarizado, desarrollado originalmente por IBM para realizar búsquedas, alterar y definir bases de datos relacionales utilizando sentencias declarativas.

2.7.7. Windows Server 2003

Es un sistema operativo de la familia Windows de la marca Microsoft para servidores que salió al mercado en el año 2003. Está basada en tecnología NT y su versión del núcleo NT es la 5.

CAPÍTULO 3

En este capítulo se van a desarrollar los módulos de Administración, Académico y DOBE del Sistema de Gestión Académica para el colegio militar Abdón Calderón No. 10, siguiendo las fases y flujos del Proceso Unificado Ágil (AUP).

3.1. Fase de Inicio

Objetivo: Identificar el alcance inicial del proyecto, proveer una arquitectura potencial para el sistema, y obtener un financiamiento inicial del proyecto y la aceptación de los stakeholders.

3.1.1. Alcance

Desarrollo e implementación de los módulos de Administración, Académico y DOBE del Sistema de Gestión Académica para el colegio militar Abdón Calderón No. 10, a ser desarrollados en un ambiente Web.

3.1.1.1. Alcance del Módulo de Administración

El módulo permitirá parametrizar las opciones de seguridad como: tiempo de vigencia de la contraseña, tamaño mínimo de contraseña, Intentos fallidos de conexión.

- El módulo permitirá llevar un registro histórico de las contraseñas generadas por los usuarios.
- El módulo permitirá gestionar módulos, formularios, perfiles y asociarlos a los perfiles para que los usuarios tengan acceso a ellos.
- El módulo permitirá gestionar perfiles, usuarios, dar acceso y permisos para los módulos a desarrollar.

3.1.1.2. Alcance del Módulo Académico

- El módulo permitirá gestionar la ubicación (país, provincia, cantón, barrio) para los estudiantes, docentes, familiares.

- El módulo permitirá parametrizar información demográfica de los estudiantes, personal, familiares como: estado civil, instrucción, tipo domicilio, tipo sangre, etc.
- El módulo permitirá realizar la gestión de los estudiantes para cada año lectivo.
- El módulo permitirá gestionar la estructura para el año lectivo como son la gestión de cursos, materias, paralelos, docentes, estudiantes, horarios de clase.
- El módulo permitirá realizar el registro de las notas de los estudiantes para cada año lectivo.
- El módulo permitirá realizar el cierre del año lectivo, una vez cerrado el año lectivo, ya no se podrán hacer modificaciones, la información solamente será de lectura.

3.1.3. Alcance del Módulo DOBE

- El módulo permitirá gestionar la ficha psicológica de los estudiantes.
- El módulo permitirá realizar seguimiento de los casos asignados a los psicólogos así como las terapias que los estudiantes deben seguir.
- El módulo permitirá visualizar reportes de los 3 módulos desarrollados.

Siguiendo el proceso, a continuación se identifican los requerimientos, los casos de uso y las entidades del negocio.

3.1.2. Especificación de Requerimientos IEEE 830

3.1.2.1. Definiciones, Acrónimos y Abreviaturas.

3.1.2.2. Definiciones

- **Administrador:** Encargado del control total del sistema, tiene acceso a todos los módulos y configuraciones del sistema, esta persona por lo general es un especialista en sistemas de información.

- **Usuario:** Persona que utilizaría la aplicación de acuerdo al nivel de autorización dado por el administrador del sistema o incluso por un súper usuario, esto se puede lograr debido a la alta parametrización del sistema.
- **Docente:** Personal encargado del pase de notas. Este usuario tendrá acceso a los cursos – paralelos en los cuales es profesor.
- **Estudiante:** es el actor principal de la aplicación, todos los procesos están en base a este usuario, es un usuario que será creado por el administrador del sistema.
- **Familiar:** Usuario que podrá llevar un control de las actividades de los estudiantes, de igual manera que los otros usuarios también tendrá un usuario y contraseña.
- **Año lectivo:** Es la configuración del inicio del año, días, horas, estructura del nuevo año lectivo.
- **Cierre de año:** cuando el año lectivo ha terminado, todos los procesos serán cerrados y ya no podrán hacer más cambios en este.
- **Horario de clases:** materias que serán dictadas en un día y hora específicos por un docente en un curso paralelo.
- **Ubicación:** proceso de asignar el país, provincia, cantón, parroquia y barrio a un estudiante, docente, familiar así como asignar uno o varios números telefónicos.
- **Círculo Familiar:** compuesto por padre, madre, hermanos o familiares cercanos al estudiante, un familiar debe ser asignado como representante del estudiante.

3.1.2.3. Acrónimo

- **ASPNET:** Es un framework para aplicaciones web desarrollado. Es usado por programadores para construir sitios web dinámicos.
- **ASPX:** Active server page, extensión de archivo de las paginas construidas por ASPNET.
- **AUP:** Agile Unified Process, proceso unificado ágil.

- **BDD** (Base de Datos)
- **DDL**: Lenguaje de definición de datos (Data Definition Language) es un lenguaje proporcionado por el sistema de gestión de base de datos.
- **DOBE**: Departamento de Orientación y Bienestar Estudiantil.
- **ERS**: Especificación de Requisitos de Software.
- **FTP**: File Transfer Protocol - Protocolo de Transferencia de Archivos
- **GUI**: Interfaz Gráfica de Usuario.
- **HTML**: Hyper Text Mark-up Language. Lenguaje de programación para páginas web.
- **HTTP**: Protocolo de transferencia de hipertexto (HyperText Transfer Protocol) es el protocolo usado en cada transacción de la Web.
- **IEEE**: (The Institute of Electrical and Electronics Engineers - Instituto de Ingenieros Eléctricos y Electrónicos) responsable de la creación de estándares.
- **IIS**: Servidor web de la familia de Microsoft.
- **RDBMS**: Relational Database Management System, Sistema Administrador de Bases de Datos Relacionales.
- **RUP**: Proceso Unificado Rational.
- **SQL**: Lenguaje de Consulta estructurado - Structured Query Language.
- **TCP/IP**: Transfer Control Protocol / Internet Protocol (Protocolo de Control de transporte y Protocolo de Internet).
- **UML**: Lenguaje Unificado de Modelamiento (Unified Modeling Language).
- **WWW**: World Wide Web mejor conocido como web.
- **XML**: Extensible Markup Language, Lenguaje de Etiquetado Extensible. Lenguaje que ofrece un formato para la descripción de datos estructurados.

3.1.2.4 ANEXO A: Especificación de Requerimientos según Norma IEEE 830.

3.1.3. Especificación de Casos de Uso

3.1.3.1. ANEXO B: Especificación de Casos de Uso y Diagramas

3.1.4. MODELOS DE DOMINIO

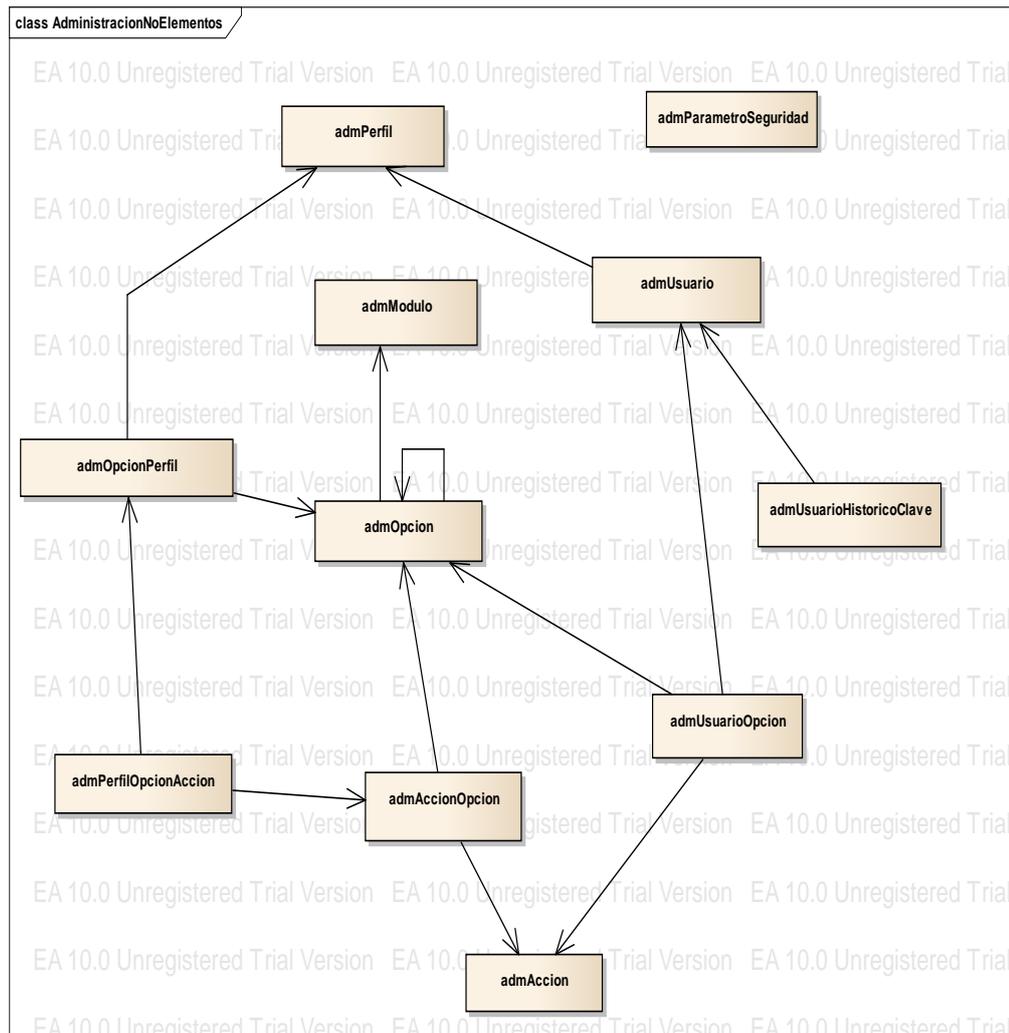


Figura 3. 1 Modelo de Dominio de Administración Módulo Académico

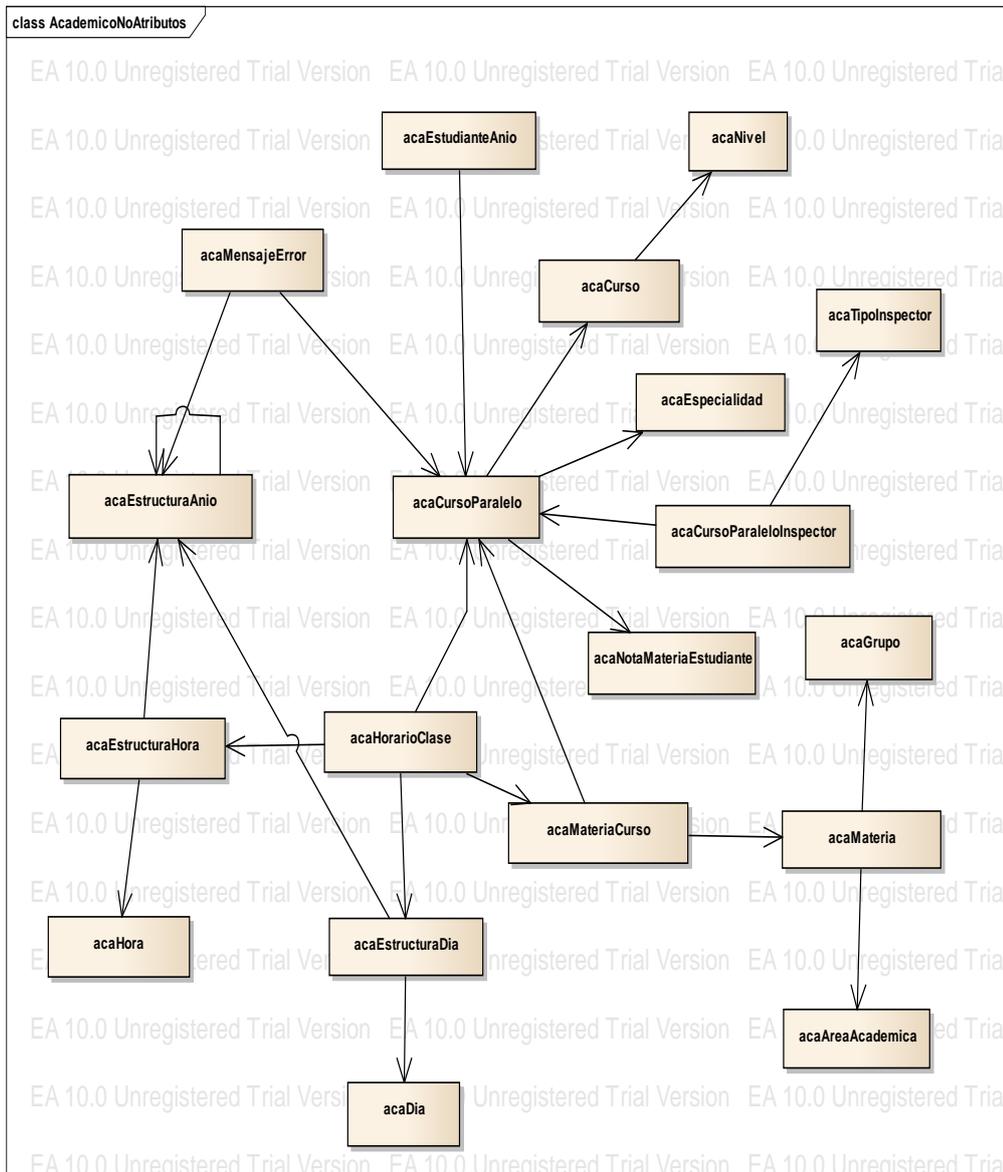


Figura 3. 2 Modelo de Dominio de Académico

3.1.4.1. Sub Módulo Secretaria

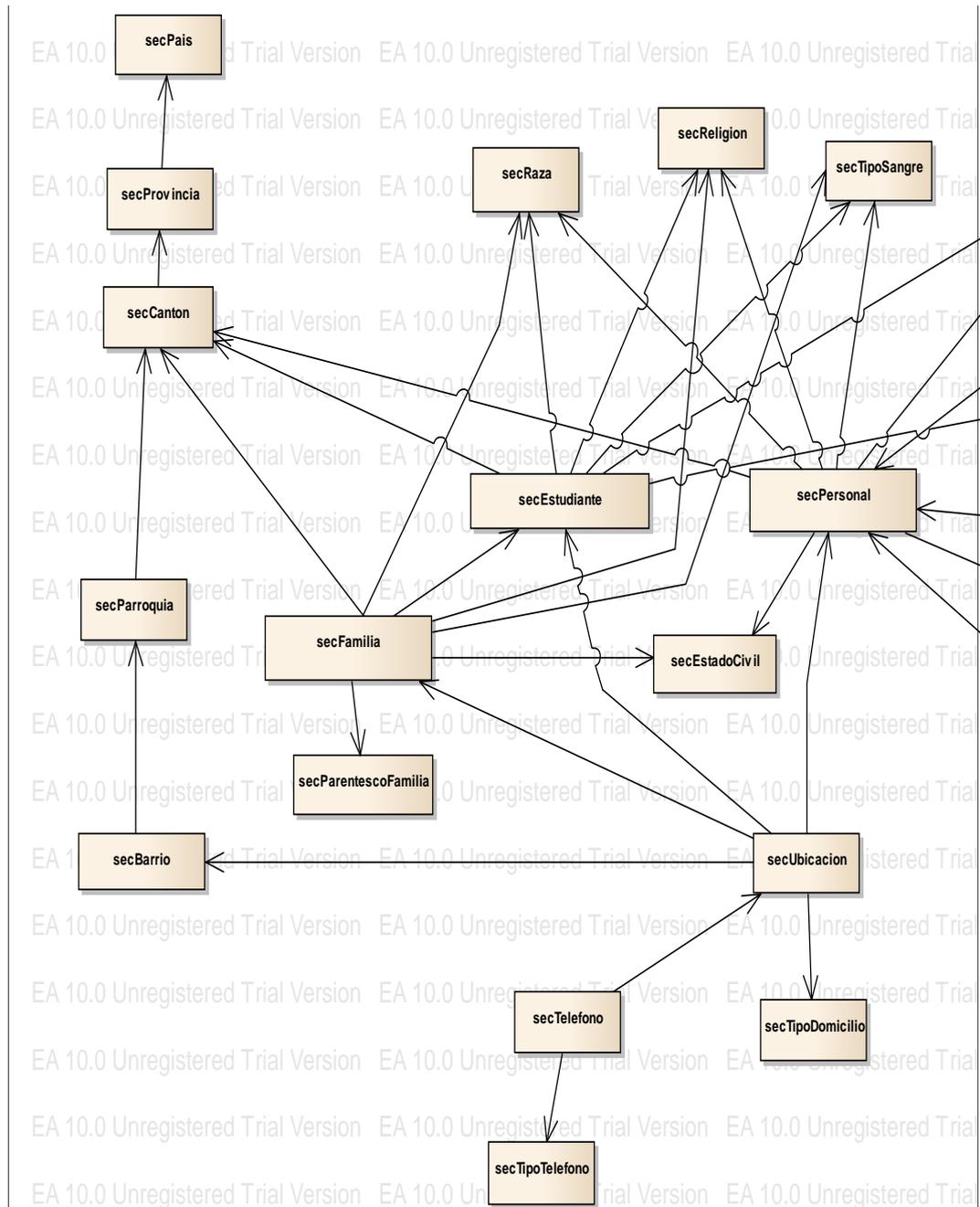


Figura 3. 3 Modelo de Dominio Sub Módulo Secretaria 1

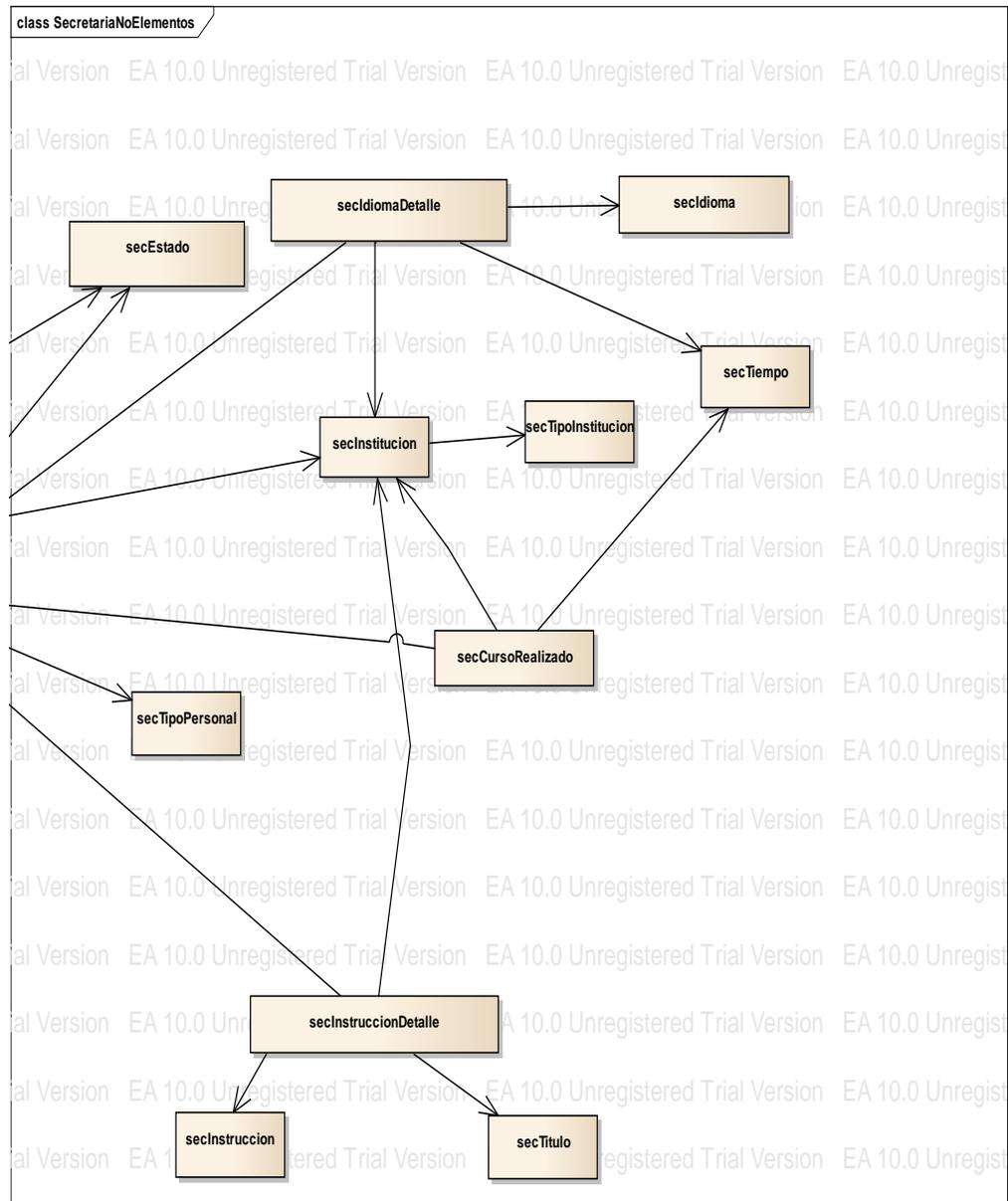


Figura 3. 4 Modelo de Dominio Sub Módulo Secretaria 2

3.1.4.2 Módulo DOBE

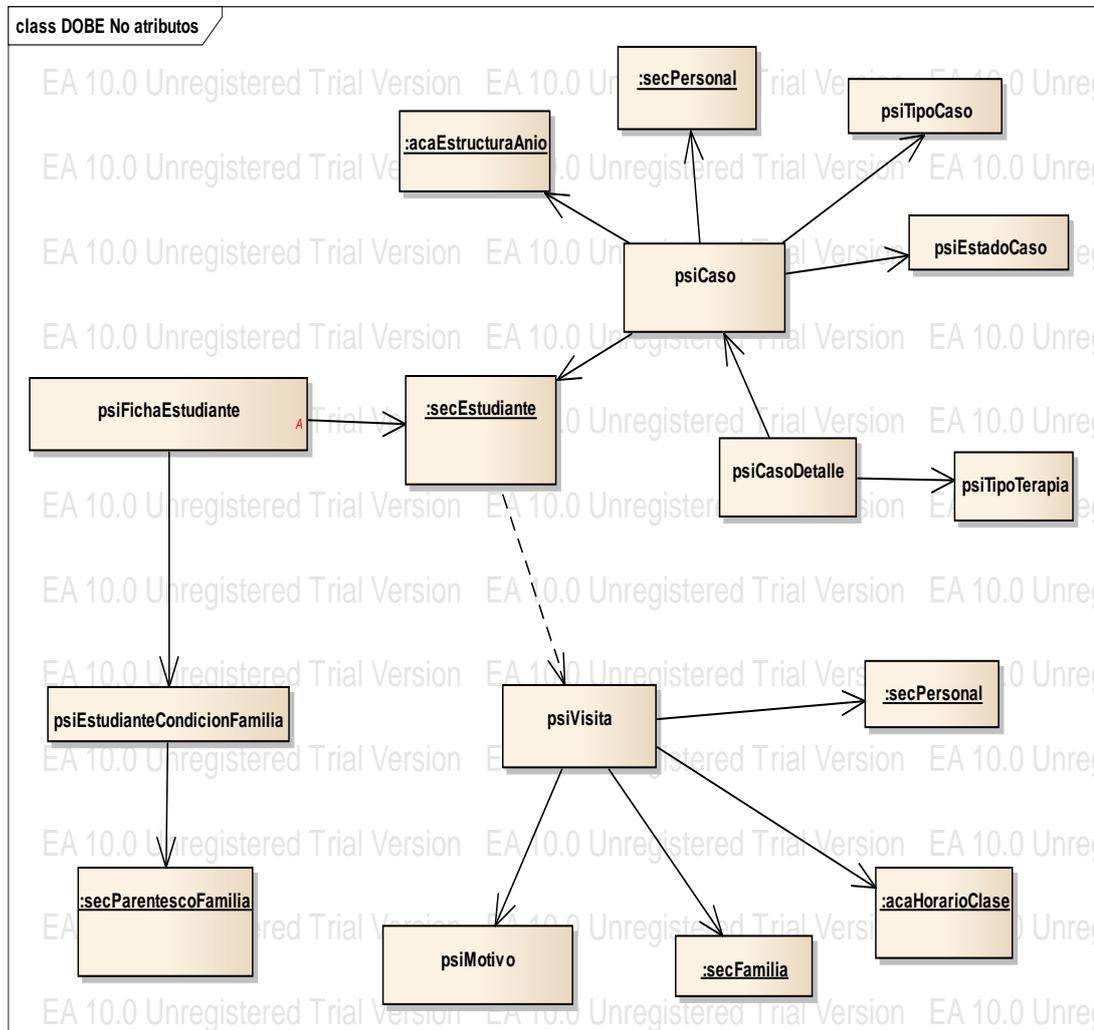


Figura 3. 5 Modelo de Dominio DOBE

3.1.5. Estimación de Costos y Gastos

Cuadro 3. 1

Estimación Costos y Gastos

Descripción	Cantidad	Tiempo	Valor Unitario	Total
Desarrolladores del proyecto	2	6 meses	0,00	\$ 0,00
Hardware				
Computador para desarrollo	2		\$ 700,00	\$ 1400,00
Total Hardware				\$ 1400,00
Licencias de operación				
Visual studio 2008 express	1		\$ 0,00	\$ 0,00
SQL Server 2008 express	1		\$ 0,00	\$ 0,00
IIS	1		\$ 0,00	\$ 0,00
Total Licencias				\$ 0,00

Continúa →

Servicio Básicos				
Luz		6 meses	\$ 40,00	\$ 240,00
Internet		6 meses	\$ 20,00	\$ 120,00
Agua		6 meses	\$ 20,00	\$ 120,00
Teléfono		6 meses	\$ 40,00	\$ 240,00
Transporte		6 meses	\$ 30,00	\$ 180,00
Varios, Suministros de Oficina	N/A		N/A	\$ 400,00
Total Servicio Básicos				\$ 1300,00
TOTAL				\$ 2700,00

3.1.6. Definición de riesgos

Tabla 3. 1

Definición de Riesgos

Riesgo	%	Impacto	Estrategia
Cambio de Rector	30	3	Aceptar
Cambio jefe académico	20	5	Aceptar
No aceptación personal administrativo	20	4	Mitigar
Suspender contrato IPs publicas	20	5	Evitar
Resistencia al cambio de la aplicación	50	3	Mitigar
Demora en la presentación de los	20		Continúa →

entregables.			
La arquitectura propuesta no va acorde a las especificaciones del diseño.	20	4	Evitar
Pobre análisis y/o diseño no satisface correctamente los requerimientos.	20	5	Evitar
Alta volatilidad y cambios en los requerimientos durante el proyecto.	30	4	Evitar
Incumplimiento en los plazos de entrega de iteraciones y versión final del producto.	20	3	Evitar

3.1.7. Matriz de Interesados

Tabla 3. 2

Matriz de Interesados

Interesado	Interés	Poder	Incide	Comunicación
Rector	5	5	Líder	Mensual
Jefe académico	5	4	Apoyo	Semanal
Personal Administrativo	4	2	Apoyo	Mensual
Docentes	3	2	Apoyo	Mensual
Psicólogos	3	2	Apoyo	Mensual
Estudiantes	3	1	Desconoce	Desconoce
Familiares	3	1	Desconoce	Desconoce
Desarrolladores	5	3	Líder	Diaria

3.1.8. ANEXO C: Diagrama Conceptual de Base de Datos.

3.1.9 ANEXO D: Diagrama Físico de Base de Datos.

3.2. Fase de Elaboración

3.2.1. Objetivo: obtener la arquitectura del sistema

Esta fase describe el diseño de la solución propuesta. Se debe justificar la elección de un patrón arquitectónico guardando una conexión con la interfaz gráfica, se identifican los patrones y estándares aceptados para el aspecto visual y la interacción grafica con el usuario.

3.2.2. Identificar arquitectura

Para el desarrollo de la solución propuesta se utilizará la arquitectura en N-Capas, debido a su diseño altamente escalable ante la incorporación de nuevos módulos y funcionalidades a futuro. Además posibilita la distribución de componentes (capas) entre varios niveles de hardware, obteniendo mayor seguridad y rendimiento ante numerosas peticiones al servidor Web.

Esta arquitectura orientada a objetos no presenta obstáculos para adaptar tanto el patrón de modelo de dominio en la capa de lógica de negocio como el patrón de repositorio en la capa de acceso a datos. La arquitectura queda dividida en cuatro capas descritas a continuación (ver Figura 3.6).

3.2.2.1. Capa de Presentación:

Esta capa integra los elementos de la interfaz gráfica y las clases con la lógica del comportamiento de las páginas para su interacción con el usuario. Involucra librerías CSS, JavaScript, Ajax, páginas maestras y ficheros ASPX y HTML.

3.2.2.2. Capa de Aplicación:

Esta capa tiene como función delegar las solicitudes de usuario provenientes de la capa previa hacia los módulos y clases correspondientes de la Capa de Lógica de Negocio,

sin involucrar la implementación en líneas de código de dicha solicitud. Asimismo actúa como fachada para futuras implementaciones de integración con otros dispositivos, plataformas y sistemas a través de aplicaciones como servicios Web.

3.2.2.3. Capa de Lógica:

Esta capa sigue la línea de trabajo de la entidad Modelo del patrón MVC. Conformada por clases cuyas funciones recaen en la implementación de la lógica de negocio atendiendo el requerimiento de usuario. Interactúa con la capa de base de datos de acuerdo con el tratamiento deseado de la información intercambiada. La codificación de la lógica de negocio sigue el patrón modelo de dominio.

3.2.2.4. Capa de Acceso a Datos:

En esta capa se ubicarán las clases DAO y librerías de conexión encargadas de administrar las operaciones CRUD (Create – Read – Update – Delete) y sentencias SQL a nivel de base de datos. La codificación de esta capa sigue el patrón repositorio.

Tabla 3. 3

Comparación requerimientos no Funcionales y Solución Propuesta

Requerimientos no funcionales	Solución propuesta
El sistema será desarrollado con una interfaz gráfica de usuario basada en controles orientados a la Web.	La capa de presentación es completamente independiente de la capa lógica y de base de datos, motivo por el cual los elementos gráficos, HTML, CSS, javascript pueden ser incorporados sin ningún problema.
El sistema será accesible desde cualquier equipo de trabajo con navegadores Web Microsoft Internet Explorer (7.0 o superior) y Mozilla Firefox (2.0 o superior).	El servidor Web IIS alojara a la aplicación Web y por el lado del cliente sólo observará código HTML compatible con los navegadores Web.
El sistema se ejecutará sobre un servidor de aplicaciones Web con sistema operativo Windows Server 2003 en adelante.	El sistema será albergado en el servidor IIS Express de libre distribución.
El sistema trabajará con el administrador de base de datos SQL Server 2008 express	El motor de base de datos SQL Server 2008 Express, es de libre uso y no requiere licencia para su uso, será el encargado de proporcionar el acceso a los datos del sistema a desarrollar.
Validación de la información	Para la validación de los datos de entrada y salida se contará con controles desarrollados bajo javascript y desde la Capa de Lógica.
Autenticación y validación	El módulo (Continúa →

	encargado de la asignación de permisos para los diferentes módulos de la aplicación.
--	--

3.2.3 Vista Lógica

La Figura 3.6 representa la vista lógica del software con las cuatro capas descritas, así como los principales componentes encargados de su funcionamiento.

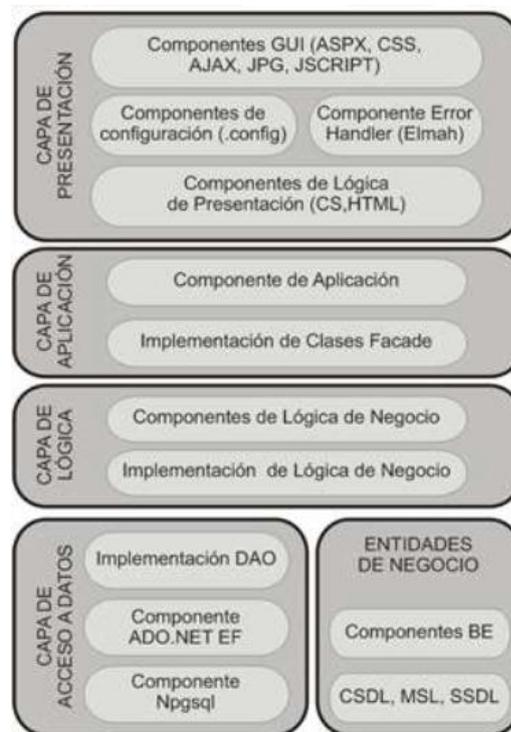


Figura 3. 6 Vista lógica del sistema

3.2.4. Vista de Despliegue

A continuación la Figura 3.7 grafica la representación de las relaciones entre los nodos físicos y su localización junto con los componentes en hardware y software

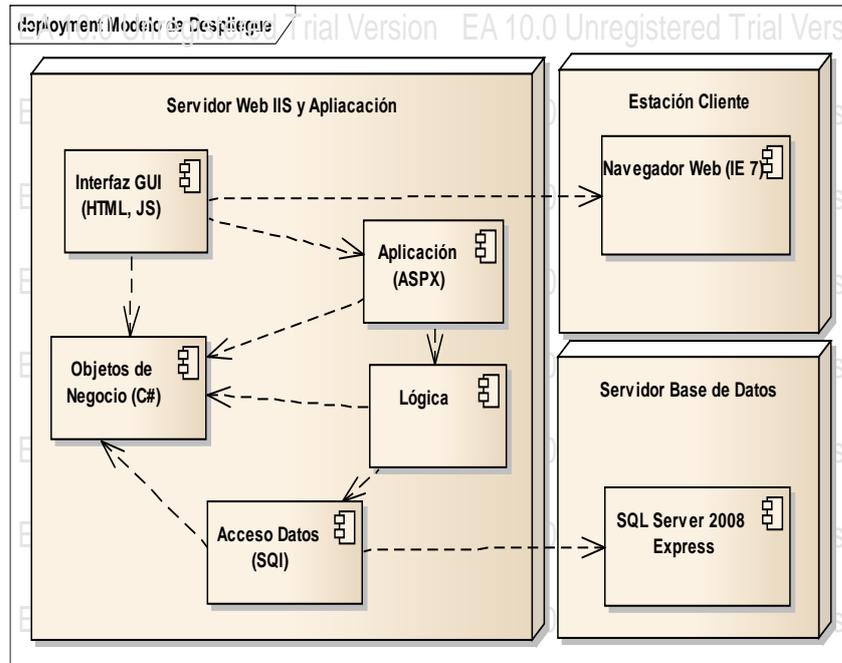


Figura 3. 7 Diagrama de despliegue

Los nodos indicados en la figura 3.7 se describen a continuación:

3.2.4.1. Estación cliente:

Este nodo representa al navegador Web de la máquina cliente, desde el cual se realiza la conexión al sistema.

3.2.4.2. Servidor Web IIS y de Aplicación:

En este nodo residen los archivos del código fuente con la lógica de negocio estructurada en capas.

3.2.4.3. Servidor de Base de datos:

Este nodo contiene el sistema administrador de base de datos. Interactúa con el nodo de servidor Web en su capa de acceso a datos.

3.2.5. Diseño de Interfaz Gráfica

La siguiente sección expone los juicios de diseño de la interfaz gráfica para la implementación de la Capa de Presentación. Adicionalmente, se detallan las restricciones en el diseño gráfico de la aplicación Web.

3.2.5.1. Estándar de Interfaz Gráfica

El patrón a seguir de las páginas que componen la aplicación es el siguiente:

Página de Inicio:

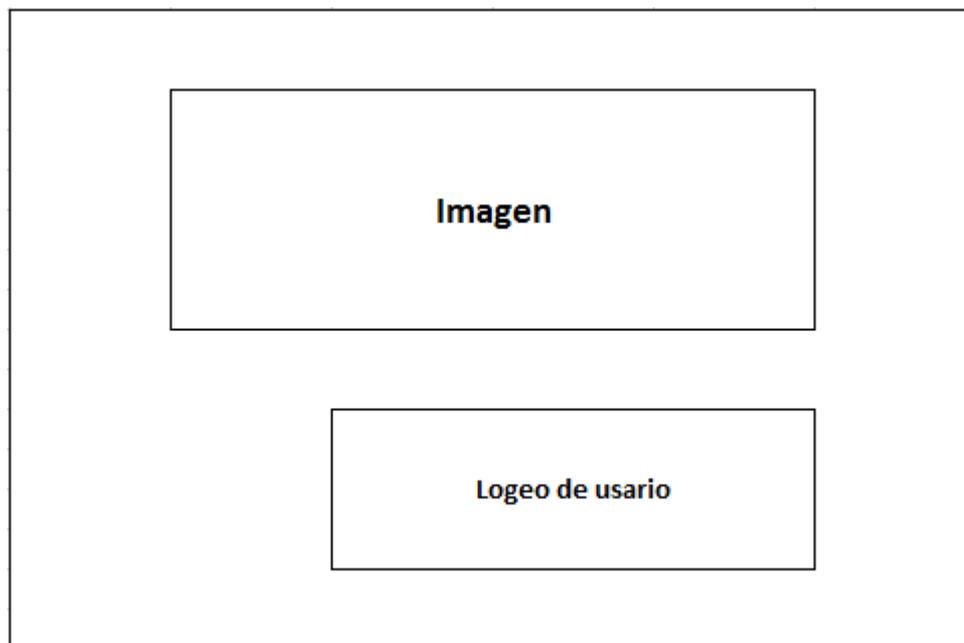


Figura 3. 8 Interfaz Gráfica Pagina Inicio

Aplicación:

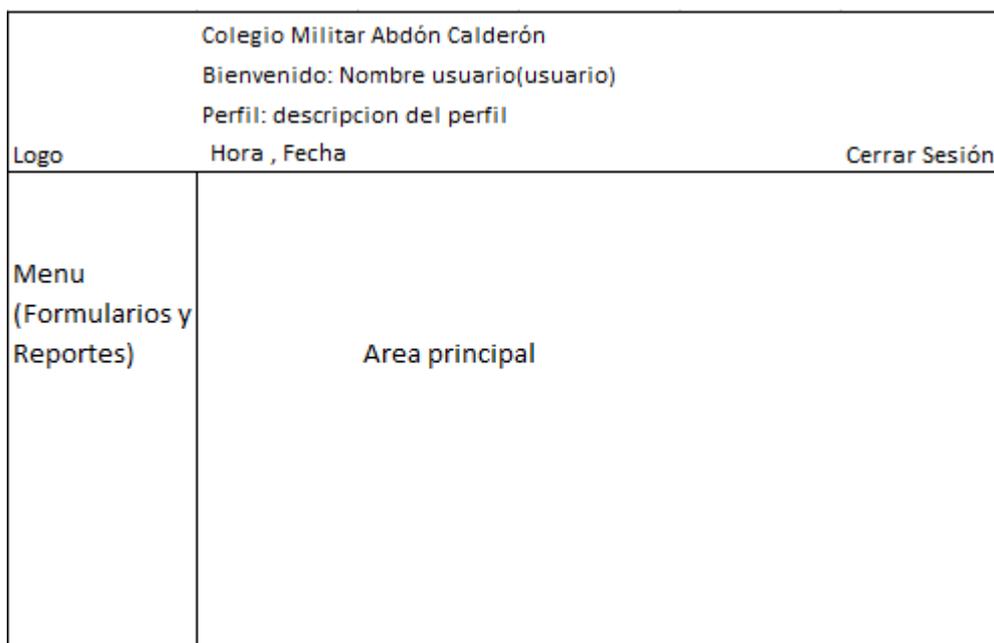


Figura 3. 9 Interfaz Gráfica Aplicación

3.2.5.2. Encabezado de página:

En la parte superior izquierda se encuentra un pequeño logotipo.

3.2.5.3. Nombre del Colegio:

En la parte superior izquierda, se mostrara el nombre del colegio.

3.2.5.4. Nombre de usuario:

En la parte superior centro, se mostrará el nombre completo del usuario.

3.2.5.5. Perfil de usuario:

En la parte superior derecha, se mostrará el perfil asociado al usuario.

3.2.5.6. Fecha de sesión:

En la parte superior izquierdo se mostrará la hora y fecha del día.

3.2.5.7. Barra de menú:

En la parte izquierda están localizados los formularios y reportes a los que los usuarios tendrán acceso.

3.2.5.8. Icono de desconexión:

En la parte superior derecha se ubicará un link de cierre de sesión de usuario, cuando el usuario presione este link será enviado a otra página.

3.2.5.9. Barras de desplazamiento:

Permite el movimiento horizontal y vertical a lo largo y ancho de la página.

3.2.5.10. Hojas de estilos en cascada (CSS):

Para el manejo de los objetos y sus propiedades que serán utilizados en la aplicación.

3.2.6. Restricciones Generales

3.2.6.1. Políticas Regulatorias

Las regulaciones dependen del nivel de permiso y configuraciones que tiene la aplicación. A nivel de permisos hay que tener en cuenta los perfiles de los usuarios. Cada usuario pertenece a un perfil y cada perfil tiene acceso a ciertos formularios, cada formulario pertenece a un módulo.

Los usuarios podrán tener acceso a más de un módulo, si el administrador considera que así sea necesario.

Las políticas regulatorias para los usuarios son:

Administrador: El administrador del sistema, tiene acceso a todos los módulos y formularios de la aplicación.

Docentes: pueden acceder a los formulario de los estudiantes para ver la información demográfica de cada uno de ellos, tienen acceso a los cursos paralelos de los son docentes para la asignación de las notas.

DOBE: pueden acceder a información de la ficha psicológica, fichas de los estudiantes, casos, seguimiento de casos, visitas domiciliarias, reportes académicos de los estudiantes.

Estudiante: pueden acceder a ver su información de la ficha psicológica, notas, ficha de datos generales, pero solo en modo lectura.

Familiar: pueden acceder a ver la información de su/sus hijo/s pero solo en modo lectura.

Los familiares que podrán acceder a ver la información son los padres o representante del estudiante.

3.2.6.2. Limitaciones del Hardware

Las características del servidor son:

- Servidor Xeon 3.06.Ghz
- 2 GB memoria RAM
- 40Gb de Disco Duro

Las PCs que utilizaran el sistema, requieren de las siguientes características:

Procesador Intel puntium III en adelante.

- 1GB de memoria RAM o superior.
- Disco duro de 20 GB o superior
- Tipo de monitor: VGA, SVG o superior.

3.2.6.3. Limitaciones de Software

- La conexión de red es el punto más importante para establecer conexión entre la aplicación y el usuario.
- Tener instalado por lo menos un browser, el más recomendado es Internet Explorer desde la versión 7.0 en adelante.

3.2.6.4. Funcionamiento en Paralelo

Hasta un total de 500 usuarios pueden estar conectados simultáneamente en la aplicación, en este lapso de tiempo podrán realizar

varias gestiones a la vez, como consultas, creación, actualizaciones, eliminaciones.

3.2.6.5. Requisitos de Lenguaje

La aplicación será desarrollada bajo la tecnología Microsoft con su línea de productos para el desarrollo como son Visual Studio 2008 con el lenguaje de programación C#, SQL Server 2008 R2 como motor de base de datos y transaction SQL con lenguaje a nivel de base de datos.

3.2.6.6. Protocolos Señalados

El protocolo TCP/IP es el principal protocolo a utilizar en el desarrollo de la aplicación.

3.2.6.7. Requisitos de Fiabilidad

El sistema debería estar disponible 24/7, pero las condiciones del cuarto de máquinas son esenciales y no se puede garantizar alta disponibilidad de la aplicación en caso de corte de energía o algún otro acontecimiento imprevisto.

El proveedor de internet debería garantizar la disponibilidad de internet pero esto sale del alcance de los desarrolladores de la aplicación.

Es responsabilidad del administrador garantizar el correcto funcionamiento del servidor, base de datos y red.

3.2.6.8. Seguridad y Condiciones de Seguridad

El módulo de administración cuenta con las características a nivel de aplicación para garantizar la seguridad, estas son:

- Tiempo máximo de conexión
- Intentos fallidos de conexión.
- Tamaño mínimo de contraseña.
- Tiempo vigencia contraseña.
- Frecuencia al mes de cambios.
- Log de contraseñas.
- Los de transacciones importantes.

3.2.7. Consideraciones finales

Las páginas no contendrán elementos dinámicos en Flash, imágenes GIF animados debido al alto consumo de memoria y procesamiento que demandan.

Se utilizará tablas HTML para contribuir con la estandarización del diseño y distribución uniforme de elementos.

Con fines de compatibilidad incrustar elementos HTML y JavaScript compatibles con los navegadores IE 6.0, Mozilla Firefox y Google Chrome. En ese sentido, los componentes gráficos Web exclusivos de la tecnología ASP.NET no formarán parte de las pantallas presente en la Capa de Presentación.

3.3. Fase de Construcción

Objetivo: implementar un software sobre una base incremental la que debe estar relacionada con los objetivos de los involucrados.

3.3.1. Estructura General

La siguiente grafica muestra la estructura general de la aplicación desarrollada.

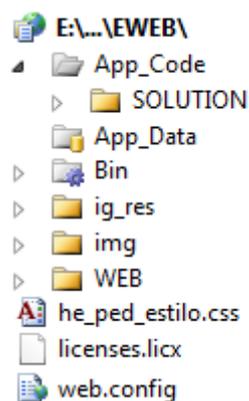


Figura 3. 10 Estructura General de la Aplicación

3.3.2. Estándares de Desarrollo

3.3.3. Estándares de Codificación

3.3.3.1. Namespace

Los nombres de los namespace serán de los módulos que se encuentran en desarrollo, el nombre debe estar en letra capital, la primera con mayúscula y las demás en minúsculas.

Los namespace de los módulos desarrollados son:

```
namespace Academico
namespace Administracion
namespace Psicologia
namespace Secretaria
```

3.3.3.2. Clases

El estándar para codificación de las clases es el siguiente:

La clase comenzará con el nombre **cla**, con el objetivo de crear diferenciación.

Después con el nombre del módulo en desarrollado, el formato del método guardará el siguiente estándar:

cla + Módulo + nombre clase

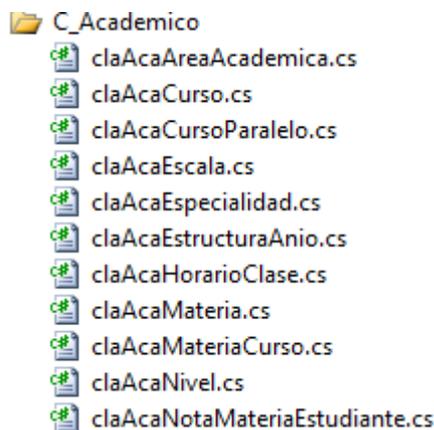


Figura 3. 11 Ejemplo de Clases

Ejemplo de clase dentro del namespace Académico

```
namespace CAcademico
{
    public class claAcaCursoParalelo
    {
    }
```

Figura 3. 12 Ejemplo Clase Namespace

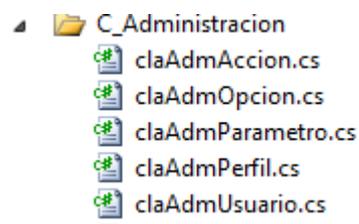


Figura 3. 13 Clases Módulo Administración

Ejemplo de clase dentro del namespace Administración

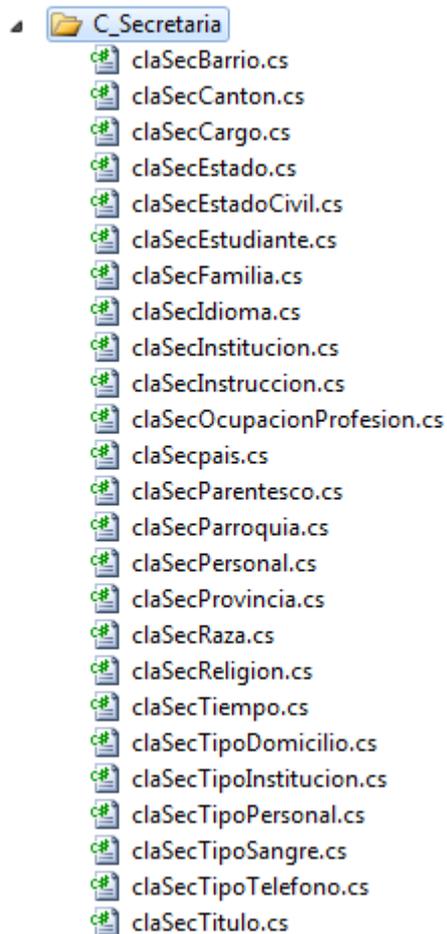


Figura 3. 14 Clases Sub Módulo Secretaria

Ejemplo de clase dentro del namespace Secretaria

```
namespace Secretaria
{
    public class claSecBarrio
    {
    }
}
```

Figura 3. 15 Clase en Namespace Secretaria

3.3.3.3. Métodos

El nombre de los métodos o funciones deberá ser un verbo en minúsculas que represente la acción que realizará; el formato del método seguirá el siguiente estándar:

fun + Clase + acción a realizar

```
public SqlDataReader funSecListadoParroquiaBarrio(Int16 parCodigo)...
public DataSet funSecListadoPaisProvinciaCantonParroquiaBarrio(...
public SqlDataAdapter funSecListadoBarrio(...
```

Figura 3. 16 Ejemplos de Métodos

3.3.3.4. Variables

El nombre de las variables debe ser corto, estos mostrarán la funcionalidad y el tipo de dato que representarán.

```
public Int16 numError;
public Int32 famCodigoNuevo;
public string edad;
public string tiempoFueraPais;
public string tiempoMurio;
public char existePadre;
public char existeMadre;
```

Figura 3. 17 Ejemplos de Variables

3.3.3.5. Comentarios

Los comentarios mostrarán una descripción del proceso que se esté realizando en las clases, métodos, algoritmos, etc.

```
// verificar si la cedula ya se encuentra ingresada
objGestionEstudiante.funSecVerificaIdentificacionInscripcion(txt_esIdentificacion.Text);
```

Figura 3. 18 Ejemplo de Comentarios

3.3.3.6. Formularios

Guardando el mismo estándar de la clases, métodos, etc. Los nombres de los formularios tendrán el siguiente formato.

Módulo + nombre formulario

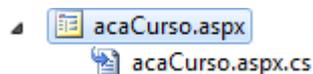


Figura 3. 19 Estándar Nombre de formularios

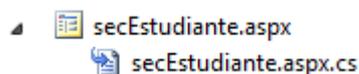


Figura 3. 20 Estándar Nombre de formularios

3.3.3.7. Archivos de validación Java Script

Guardando el mismo estándar de la clases, métodos, etc. Los nombres de los archivos de validación javascript tendrán el siguiente formato.

Js + Módulo + nombre archivo



3.3.4. Estándares de Base de Datos

3.3.4.1. Tablas

Guardando el mismo estándar de la clases, métodos, etc. Los nombres de las tablas tendrán el siguiente formato.

Módulo + nombre tabla

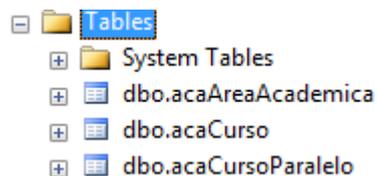


Figura 3. 21 Estándar nombre Tablas

3.3.4.2. Campos tablas

El estándar para los campos de las tablas será el siguiente:

Si el nombre de la tabla es simple, se tomarán las tres primeras letras del nombre de la tabla, como ejemplo se tomará la tabla acaCurso:

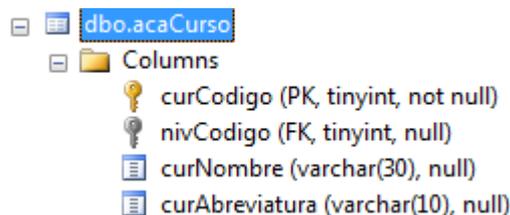


Figura 3. 22 Estándar campos de la tabla

Si el nombre de la tabla tiene más de una palabra, el nombre del campo se formará con las tres primeras letras del primer nombre seguido por la primera letra del segundo nombre, como ejemplo se tomará la tabla acaCursoParalelo.

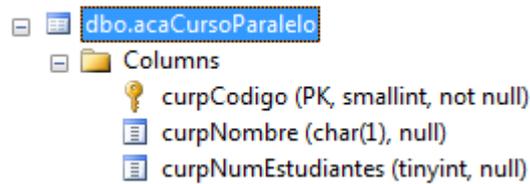


Figura 3. 23 Estándar Campos de tabla

3.3.4.3. Procedimientos almacenados

El estándar de los procedimientos almacenados tendrá el siguiente formato.

Sp + módulo + acción a realizar

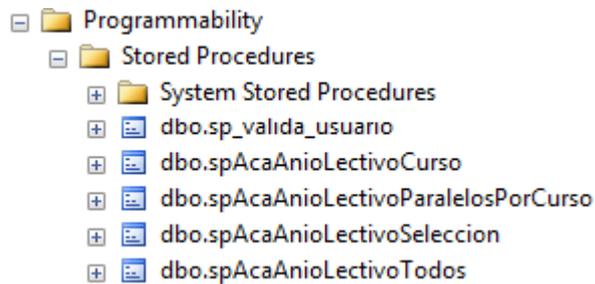


Figura 3. 24 Estándar Procedimientos Almacenados

3.3.4.4. Funciones

El estándar de las funciones tendrá el siguiente formato.

fun + módulo + acción a realizar

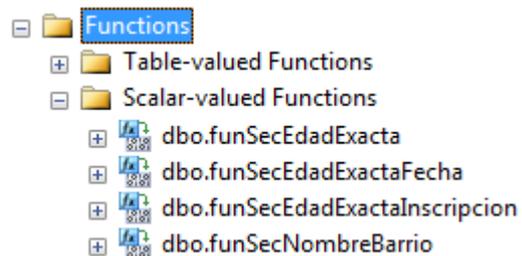


Figura 3. 25 Estándar Funciones de Base de Datos

3.3.5. Desarrollo de interfaces de Usuario

Continuando con el proceso de desarrollo se mostrarán algunas interfaces de usuario en modo de diseño.

3.3.5.1. Diseño formulario de Usuario

WEB/SIT_1_0/Com/...I/admUsuario.aspx

body

ScriptManager - ScriptManager1

Nuevo Guardar Regresar Permiso Adicional

[IblMensaje]

Username:

Tipo usuario: [IblTipoUsuario]

Nombres: [IblNombres]

Perfil: Unbound

Estado: ACTIVO

Figura 3. 26 Diseño del formulario de Usuario

3.3.5.2. Diseño formulario de Estudiante

WEB/SIT_1_0/Com/S...ecEstudiante.aspx WEB/SIT_1_0/Com/S...Estudiante.aspx.cs WEB/SIT_1_0/Com/...I/admUsuario.aspx

ScriptManager - ScriptManager1
asp:UpdatePanel#UpdatePanel1

Guardar ir a buscar [Ibl_formulario]

[Ibl_mensaje]

Curso inscripción: Unbound

DATOS GENERALES [IblEstudianteInscrito]

Codigo: 0

*Apellido paterno:

*Nombres:

*Fecha Nacimiento:

*Lugar Nacimiento:

Tipo Identificación: CEDULA

*Sexo: Femenino Masculino

*Tipo Sangre: Unbound

Raza: Unbound

Estado: Unbound

Número calzado:

Talla(cm):

Color cabello:

REGISTRO DE INFORMACIÓN DEL ASPIRANTE - [LbAnioLectivo]

NOTA: Los campos marcados con * son obligatorios.

*Apellido Materno:

Nombre pila:

Edad: [Ibl_es_edad]

*Identificación: No validar identificación

Institución viene: Unbound

Religion: Unbound

Peso Corporal (libras):

Color ojos:

Figura 3. 27 Diseño del formulario de Estudiante

3.3.5.3. Diseño formulario de Ficha Psicológica

ScriptManager - ScriptManager1

[lbl_formulario]

Guardar

[lbl_mensaje]

CONDICIONES FAMILIARES

*El estudiante vive con: Unbound

*Al llegar a casa el estudiante tiene compañía?: SI NO

*Las relación con su madre es ? :

*Las relación con sus hermanos es ? :

*Cuantas horas pasa solo?:

*Las relación con su padre es ? :

*Tipo de Hogar:

CONDICIONES DE CRECIMIENTO

*Embarazo: DESEADO NO DESEADO

*Enfermedad embarazo:

*Edad gestacional:

*Lugar:

*Tipo de parto:

*Problemas de salud antes de cumplir un año de edad?: SI NO

*Alimentación: MATERNA ARTIFICIAL MIXTA

*Gateo: SI NO

Figura 3. 28 Diseño del formulario de Ficha Psicología

3.3.5.4 Diseño formulario de Estudiante Nota

WEB/SIT_1_0/Com/...tudianteNota.aspx

ScriptManager - ScriptManager1

[lb_mensaje_2][lb_mensaje]

[lb_formulario]

Guardar Seleccione la materia:

Unbound

Curso-Paralelo:[lbCurpNombre]; perteneciente al:[lbAnioLectivo]

Listado Curso Paralelo

- Root
 - Parent 1
 - Leaf 1
 - Leaf 2
 - Parent 2
 - Leaf 1
 - Leaf 2

Estudiantes Curso Paralelo

Column0	Column1	Column2
abc	abc	abc

Seleccione el periodo academico

- Root
 - Parent 1
 - Leaf 1
 - Leaf 2
 - Parent 2
 - Leaf 1
 - Leaf 2

Figura 3. 29 Diseño del formulario de Estudiante Nota

3.4. Fase de Transición

3.4.1. Pruebas

Se detallarán los procedimientos de las pruebas durante el proceso de desarrollo del software, los tipos de pruebas seleccionadas y aplicación de cada uno de estos, las pruebas del sistema deben cumplir varios objetivos como: recuperación, seguridades, resistencia, rendimiento del sistema.

3.4.1.1. Pruebas de Unidad

Se realizarán las pruebas de unidad, verificarán el desempeño de los métodos de los módulos del sistema.

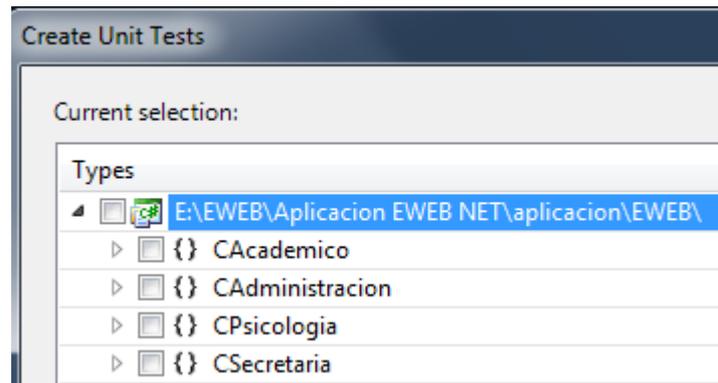


Figura 3. 30 Módulos de Prueba

3.4.1.2. ANEXO E: Pruebas Unitarias de Software.

3.4.2. Pruebas de Stress

El objetivo de las pruebas de stress es validar el funcionamiento del aplicativo bajo ciertas condiciones de trabajo, en las cuales se determina la respuesta del sistema ante las solicitudes de los usuarios, la red, la memoria del servidor.

Realizar pruebas de stress o de esfuerzo es muy recomendable porque es una simulación que tendrá el software en la vida real, en la cual se pueden encontrar errores que en ambientes de desarrollo no son fáciles de ubicar.

Para las pruebas de stress se utilizará el software de libre distribución WebServer Stress Tool 8 con las siguientes configuraciones:

Tipo de Test: por Clicks

Número de usuarios concurrentes: 200

Tiempo de espera entre solicitudes: 10 segundos

Select Test Type and Number of Users

Test Type

CLICKS Run Test with constant load until each users has generated a specified number of clicks

TIME Run Test with constant load for a specified time

RAMP Run Test with increasing load for a specified time

Run until Clicks Per User

User Simulation

Number Of Users

Click Delay Seconds Random Click Delay Use "per URL" click delay

Estimated load for 200 users dicking a link every 10 seconds:
~1.200 pageviews/minute (~72.000 pageviews/hour)

Project/Scenario Comments, Operator

Figura 3. 31 Configuración pruebas de stress

Resultado de prueba de stress

View Logfile Results

Logfiles Results per User (Complete Test) Results per URL (Complete Test)

* Summary Log *

Detailed Log.txt (0 MB)
User 00001.txt (0 MB)
User 00002.txt (0 MB)
User 00003.txt (0 MB)
User 00004.txt (0 MB)
User 00005.txt (0 MB)
User 00006.txt (0 MB)
User 00007.txt (0 MB)
User 00008.txt (0 MB)
User 00009.txt (0 MB)
User 00010.txt (0 MB)
User 00011.txt (0 MB)
User 00012.txt (0 MB)
User 00013.txt (0 MB)
User 00014.txt (0 MB)
User 00015.txt (0 MB)
User 00016.txt (0 MB)
User 00017.txt (0 MB)
User 00018.txt (0 MB)
User 00019.txt (0 MB)
User 00020.txt (0 MB)
User 00021.txt (0 MB)
User 00022.txt (0 MB)
User 00023.txt (0 MB)
User 00024.txt (0 MB)
User 00025.txt (0 MB)
User 00026.txt (0 MB)
User 00027.txt (0 MB)
User 00028.txt (0 MB)
User 00029.txt (0 MB)
User 00030.txt (0 MB)
User 00031.txt (0 MB)
User 00032.txt (0 MB)
User 00033.txt (0 MB)
User 00034.txt (0 MB)
User 00035.txt (0 MB)
User 00036.txt (0 MB)
User 00037.txt (0 MB)

Summary Log

```

3
4 Test run on 01/03/2015 11:38:39
5
6 ** Project and Scenario Comments, Operator **
7
8
9
10 Results of period #1 (from 2 sec to 12 sec):
11 *****
12 Completed Clicks: 298 with 0 Errors (=0,00%)
13 Average Click Time for 300 Users: 100 ms
14 Successful clicks per Second: 29,37 (equals 105.743,32 Clicks per Hour)
15
16 Results of complete test
17 *****
18
19 ** Results per URL for complete test **
20
21 URL #1 (): Average Click Time 74 ms, 100 Clicks, 0 Errors
22 URL #2 (): Average Click Time 109 ms, 99 Clicks, 0 Errors
23 URL #3 (): Average Click Time 117 ms, 99 Clicks, 0 Errors
24
25 Total Number of Clicks: 298 (0 Errors)
26 Average Click Time of all URLs: 100 ms
27
28
29 !! Glossary:
30 !! Click: A simulated mouse click of a user sending a request (one of the URLs from the URL list) to the server and immediately r
31 !! Request: A HTTP request sent to the server regardless of an answer.
32 !! Hit: A completed HTTP request (i.e. sent to the server and answered completely). Hits can be the PAGE request of a "click" o
33 !! Time for DNS: Time to resolve a URL's domain name using the client system's current DNS server.
34 !! Time to connect: Time to set up a connection to the server.
35 !! Time to first byte (TFB): Time between initiating a request and receiving the first byte of data from the server.
36 !! Click Time: The time a user had to wait until his "click" was finished (including redirections/frames/images etc.).
37 !! User Bandwidth: The bandwidth a user was able to achieve.
38 !! Sent Requests: Number of requests sent to the server during a period.
39 !! Received Requests: Number of answers received from the server during a period.

```

Figura 3. 32 Resultado Pruebas de Stress

Logfiles	Results per User (Complete Test)				Results per URL (Complete Test)		
User No.	Clicks	Hits	Errors	Avg. Click Time [ms]	Bytes	kbit/s	
1	1	1	0	49	12.620	2.049,58	
2	1	1	0	172	1.034	48,10	
3	1	1	0	361	1.062	23,57	
4	1	1	0	205	12.620	493,58	
5	1	1	0	58	1.034	141,46	
6	1	1	0	164	1.062	51,85	
7	1	1	0	47	12.620	2.150,27	
8	1	1	0	63	1.034	131,99	
9	1	1	0	360	1.062	23,60	
10	1	1	0	57	12.620	1.776,61	
11	1	1	0	361	1.034	22,94	
12	1	1	0	167	1.062	50,82	
13	1	1	0	168	12.620	599,76	
14	1	1	0	402	1.034	20,58	
15	1	1	0	318	1.062	26,68	

Figura 3. 33 Resultado pruebas de Stress Usuarios

Logfiles	Results per User (Complete Test)				Results per URL (Complete Test)		
User No.	Clicks	Hits	Errors	Avg. Click Time [ms]	Bytes	kbit/s	Cookies
290	1	1	0	72	1.034	115,14	
291	1	1	0	46	1.062	185,19	
292	1	1	0	39	12.620	2.616,41	
293	1	1	0	31	1.034	263,61	
294	1	1	0	51	1.062	166,15	
295	1	1	0	40	12.620	2.531,02	
296	1	1	0	46	1.034	181,54	
297	1	1	0	33	1.062	256,11	
298	1	1	0	19	12.620	5.253,12	
299	1	1	0	31	1.034	267,59	
300	1	1	0	53	1.062	160,05	

Figura 3. 34 Resultado pruebas de Stress Usuarios

3.4.3. Instalación del Sistema

Terminada las pruebas del sistema se procederá a la instalación y puesta en producción de los módulos de Administración, Académico y Psicología.

3.4.3.1. Puesta en Producción

3.4.3.2. Compilación de la Solución

En el IDE de desarrollo de Visual Studio, realizamos los siguientes pasos:

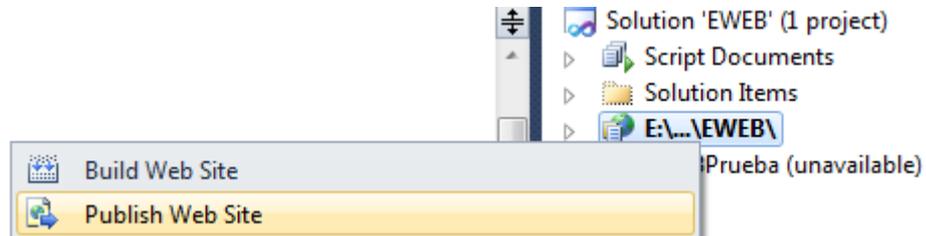


Figura 3. 35 Publicar Solución

3.4.3.3. Configuración y generación de código

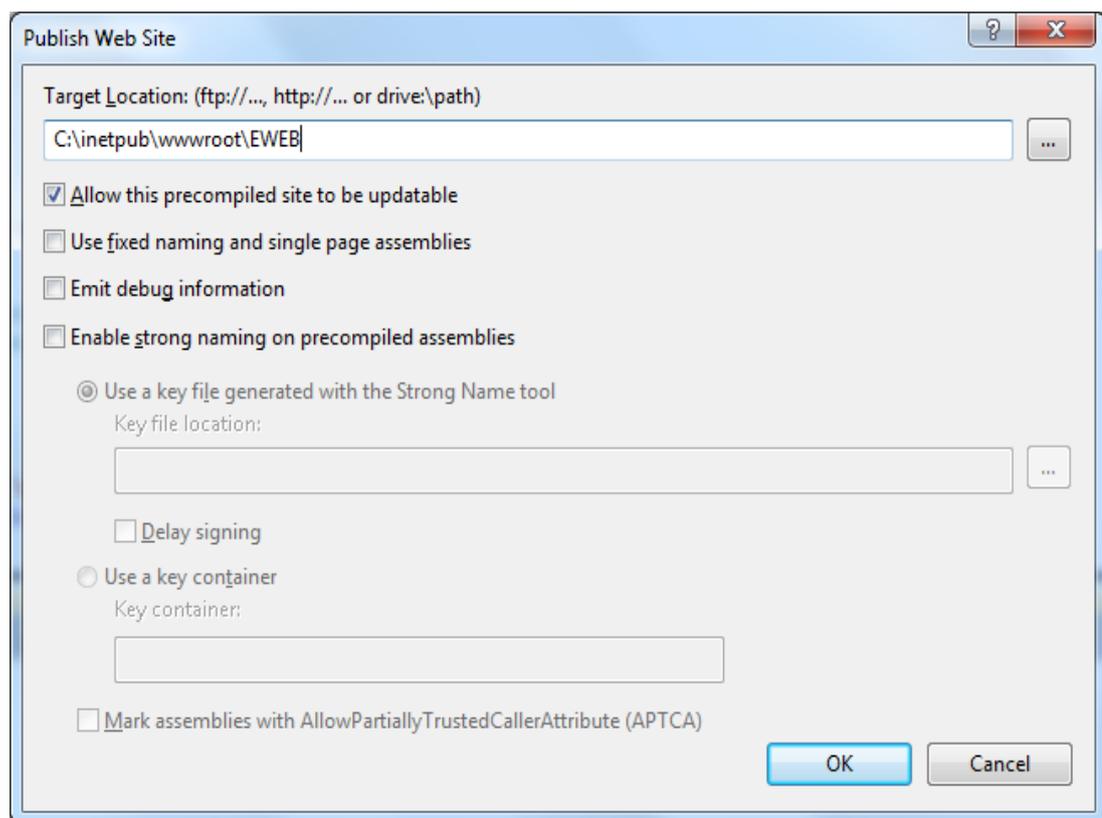


Figura 3. 36 Configurar Carpeta Solución

3.4.3.4. Configuración del Servidor de Producción(IIS)

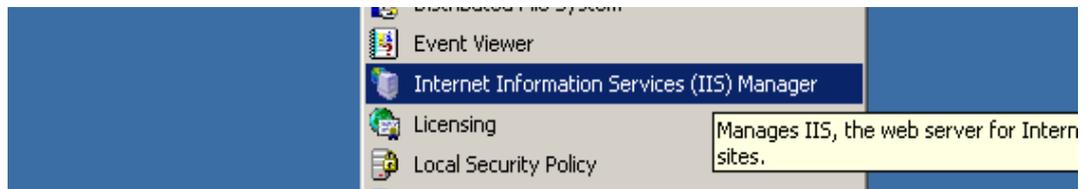


Figura 3. 37 Configurar Servidor Web IIS

3.4.3.5. Creación del Sitio Web EWEB

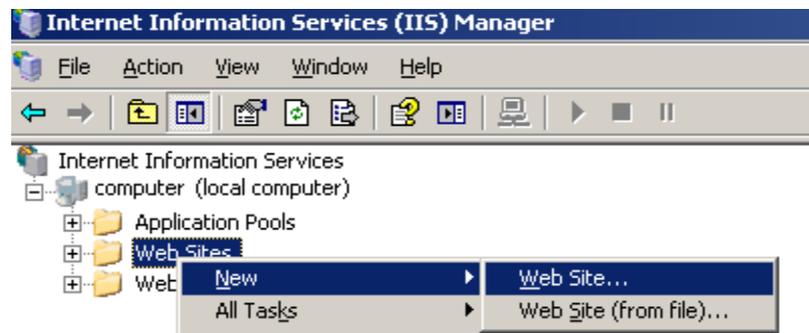


Figura 3. 38 Crear Nombre Sitio Web

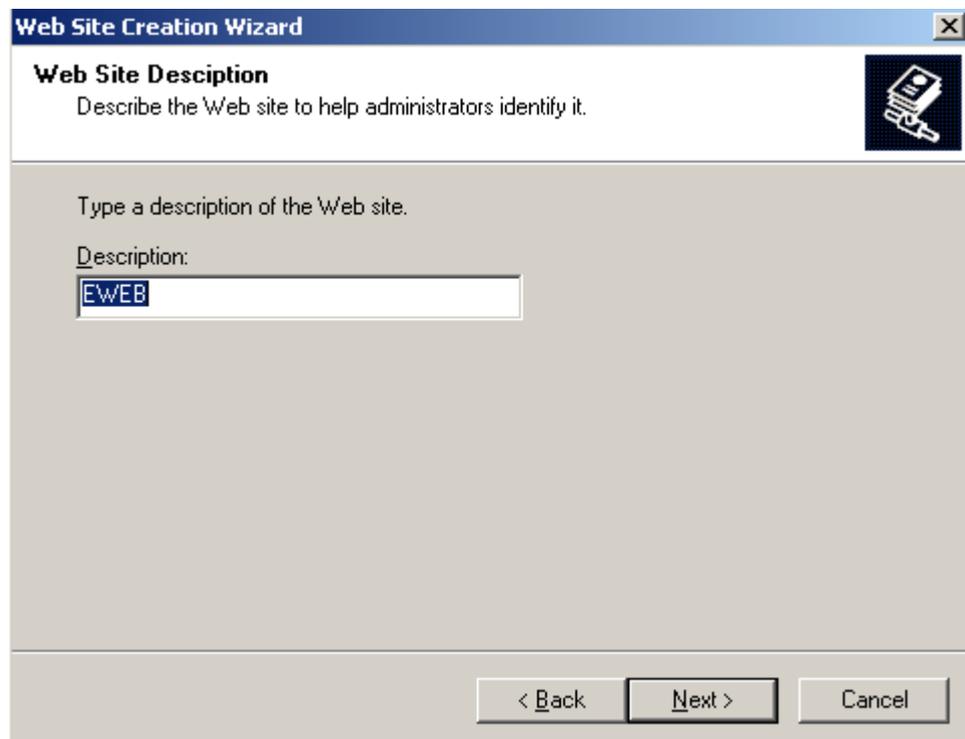
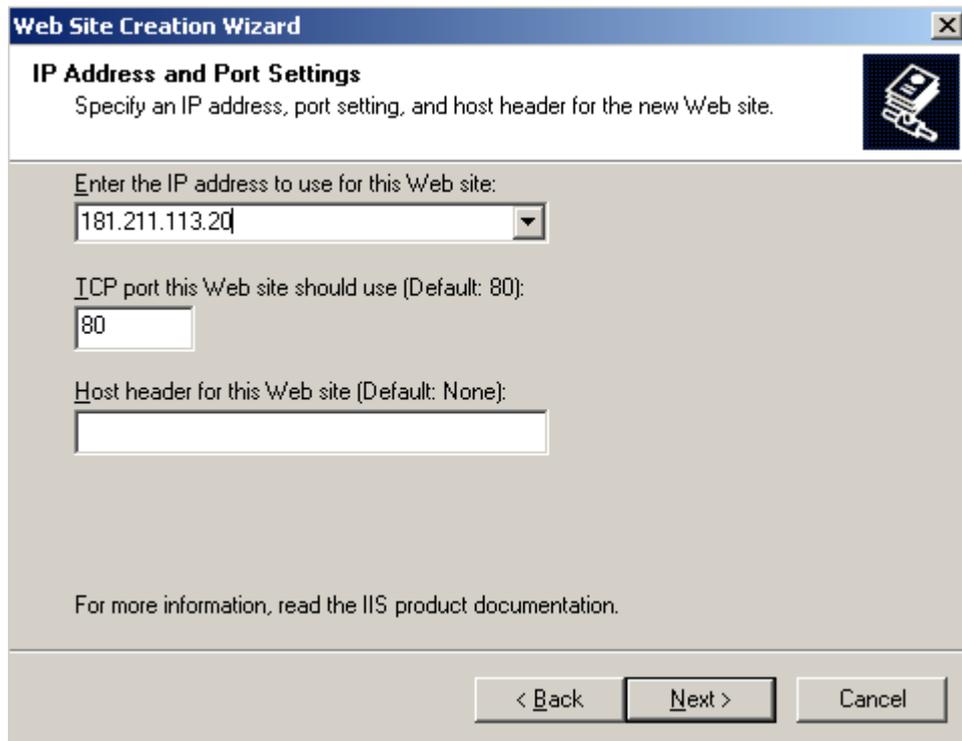


Figura 3. 39 Crear Nombre Sitio Web

3.4.3.6. Asignación de la IP Pública



Web Site Creation Wizard

IP Address and Port Settings
Specify an IP address, port setting, and host header for the new Web site.

Enter the IP address to use for this Web site:

TCP port this Web site should use (Default: 80):

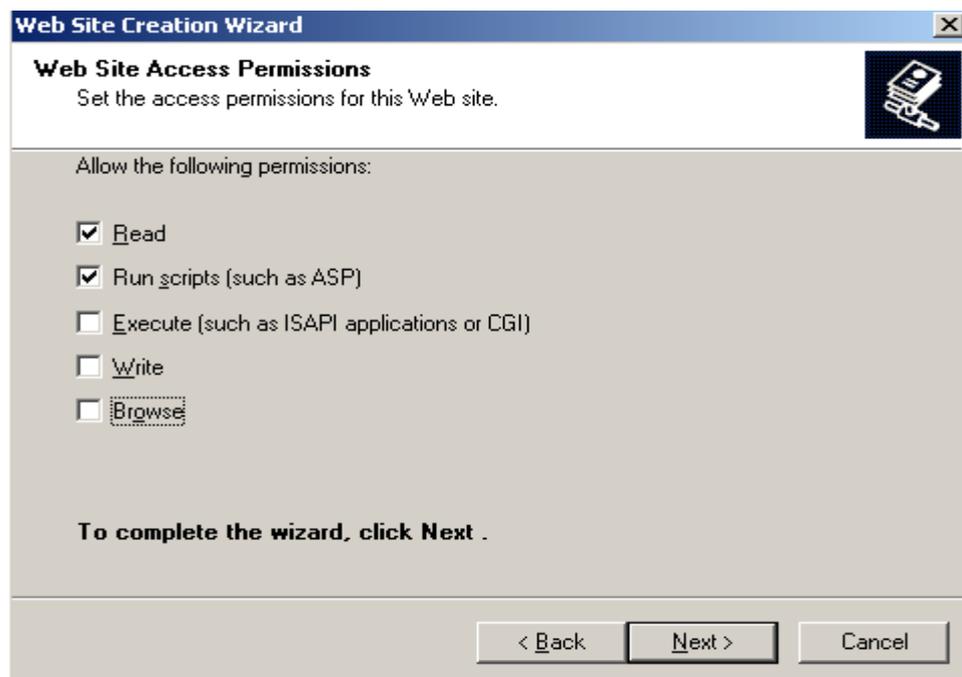
Host header for this Web site (Default: None):

For more information, read the IIS product documentation.

< Back Next > Cancel

Figura 3. 40 Asignar IP Pública Sitio Web

3.4.3.7. Asignación de permisos



Web Site Creation Wizard

Web Site Access Permissions
Set the access permissions for this Web site.

Allow the following permissions:

- R**ead
- R**un s_cripts (such as ASP)
- E**xecute (such as ISAPI applications or CGI)
- W**rite
- B**rowse

To complete the wizard, click **Next**.

< Back Next > Cancel

Figura 3. 41 Asignar Permisos Sitio Web

3.4.3.8. Sitio Web en producción

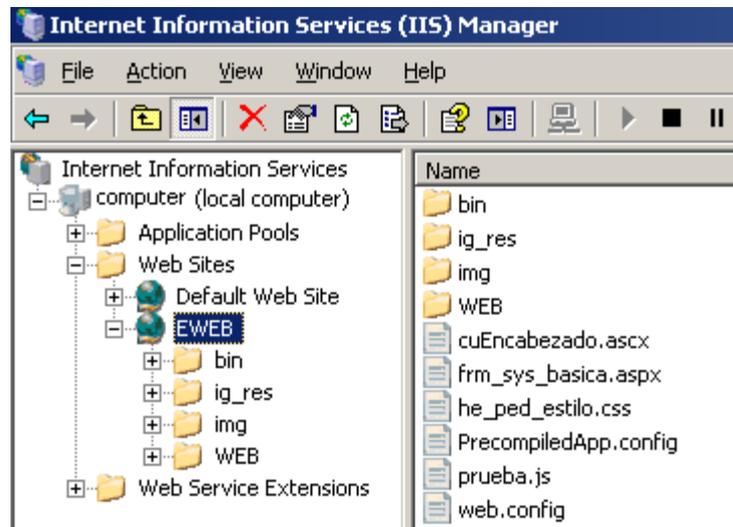


Figura 3. 42 Vista Sitio Web

3.4.3.9. Pantalla inicial de sistema



Figura 3. 43 Formulario Ingreso a la Aplicación

3.4.3.10 Búsqueda de estudiantes

Resultado búsqueda estudiante
haga doble clic sobre el registro que desea visualizar

Edit Column	Delete Column	Codigo	Apellido Paterno	Apellido Materno	Nombres	Nombre Pila	Identificación	Sexo	T.Sangre	Raza	Religión	Estado
Edit	Delete	109	PEREZ	ALMEIDA	KEVIN ALEJANDRO		1718323809	MASCULINO	ORH+			RETRADO
Edit	Delete	7084	PEREZ	ALMEIDA	ALAN STEVEN		1724466097	MASCULINO	ORH+			ACTIVO

Figura 3. 44 Formulario de Búsqueda de Estudiantes

3.4.3.11. Creación de estudiantes

Guardar ir a buscar

DATOS GENERALES

Codigo: 0

*Apellido paterno:

*Nombres:

*Fecha Nacimiento:

*Lugar Nacimiento:

Tipo Identificación: CEDULA

*Sexo: Femenino Masculino

*Tipo Sangre:

Raza:

Estado: ACTIVO

Número calzado:

Talla(cm):

Color cabello:

*Apellido Materno:

Nombre pila:

Edad:

*Identificación: No validar identificación

Institución viene:

Religion:

Peso Corporal(libras):

Color ojos:

TELEFONOS
Table control

UBICABILIDAD

Figura 3. 45 Formulario Estudiante

3.4.3.12. Catálogo General

The screenshot displays the 'Formulario Estado Civil' interface. At the top, the header includes the logo 'web educación', the institution name 'COLEGIO MILITAR ABDON CALDERON 10', the user name 'Bienvenido :ABRIL DORA CATALINA(ocarate)', the profile 'Perfil: ADMINISTRADOR', the time '17:17:35', the date 'domingo, 01 de marzo de 2015', and a 'Cerrar Sesión' link.

The navigation menu on the left lists:

- Pantallas
- Reportes
- ADMINISTRACION
- SECRETARIA
 - Busqueda/Creación Estu
 - Catálogo General
 - Ubicabilidad
 - Gestión Personal
 - Catálogo Personal
 - Busqueda Personal
- INSPECCION
- ACADEMICO

The main content area features a 'Formulario Estado Civil' with a 'Nombre:' input field and three buttons: 'Nuevo', 'Guardar', and 'Eliminar'. Below the form is a table with the following data:

Edit Column	Delete Column	Codigo	Estado Civil
Edit	Delete	1	CASADO
Edit	Delete	4	DIVORCIADO
Edit	Delete	2	SOLTERO
Edit	Delete	5	UNION LIBRE
Edit	Delete	3	VIUDO

Figura 3. 46 Catálogo General

CAPÍTULO 4

4.1. Conclusiones y Recomendaciones

4.1.1. Conclusiones

- La Institución hasta la fecha había estado trabajando con una aplicación antigua y que generaba muchos problemas. Con la puesta en producción de los módulos de Administración, Académico y Secretaria ayudará en gran manera a realizar de manera más eficiente las tareas de los docentes, personal administrativo, estudiantes y padres de familia.
- Se ha dado el primer paso para que en un futuro se pueda integrar todos los módulos de la institución, pues el módulo de Administración fue diseñado y desarrollado con este propósito.
- Tanto los padres de familia como los estudiantes podrán seguir en línea las actividades de los estudiantes, este es un gran paso pues la aplicación está alojada en una IP Pública, con lo cual esta podrá ser consultada desde cualquier PC que disponga de internet. Así, acercamos la institución a los padres y estudiantes.
- Utilizar las herramientas Visual Studio como IDE de desarrollo y SQL Server Express como motor de base de datos, facilitó desarrollar, probar e implementar los módulos en un ambiente integrado.
- Los módulos desarrollados cumplen con las especificaciones de requerimientos solicitadas por el cliente, con las cuales se asegura la integridad de la información, disponibilidad y confiabilidad de los datos en un ambiente con las configuraciones adecuadas de seguridad del Servidor y de los módulos desarrollados.

4.1.2 Recomendaciones

- Para obtener todas las ventajas de los módulos desarrollados, las autoridades son responsables de socializar y promover el uso de la aplicación, se debe educar de una manera adecuada al usuario final para que maneje de manera adecuada su usuario y contraseña.
- Las capacitaciones deben ser tanto al personal administrativo, docente, estudiantes y padres de familia, todos son parte en la correcta utilización de la aplicación.
- El administrador de la aplicación deberá realizar un plan de mantenimiento, es decir, borrado del log de la base de datos, reconstrucción de los índices de las tablas, esto, con el fin de mantener adecuados tiempos en el procesamiento de las transacciones.
- El servidor donde fue instalada la aplicación, de preferencia debe ser únicamente para el uso de este, instalar otros programas o utilizarlo con otro fin podría afectar al rendimiento de la aplicación. Además, es una ventana abierta para que se instalen virus, comprometiendo la integridad de la aplicación.
- El manejo de las seguridades de la aplicación y de la base de datos debe ser exclusivamente del administrador, no considerar este punto puede generar muchos problemas en el servidor, servidor Web y base de datos. Se recomienda crear políticas de acceso a los servidores de la institución.

Bibliografía

- Alegsa. (2013). Obtenido de http://www.alegsa.com.ar/Respuesta/ventajas_y_desventajas_de_las_aplicaciones_web.htm
- Aplicaciones en Capas. (2013). Obtenido de <http://oness.sourceforge.net/proyecto/html/ch03s02.html>
- Internet Ya. (2013). Obtenido de <http://www.internetya.co/ventajas-y-beneficios-de-las-aplicaciones-web/>
- Manual de Umbrello UML Modeller. (Junio de 2013). Obtenido de <http://docs.kde.org/stable/es/kdesdk/umbrello/uml-elements.html>
- Marco de Desarrollo de la Junta Andalucía. (2013). Obtenido de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/212>
- Wikipedia. (2013). Obtenido de <https://es.wikipedia.org/wiki/Wikipedia:Portada>
- Ambler, S. W. (13 de Mayo de 2006). Introducción al Agile UP. Obtenido de <http://www.cc.una.ac.cr/AUP/html/overview.html>
- Booch, G., Jacobson, I., & Rumbaugh, J. (1999). El Lenguaje Unificado de Modelado. En A. I. Wesley. Madrid.
- Booch, G., Jacobson, I., & Rumbaugh, J. (2000). El Proceso Unificado de Desarrollo de Software. En A. Wesley. Pearson Educación S.A. España.
- Flores Ervin, C. L. (6 de Septiembre de 2013). PROCESO UNIFICADO ÁGIL (AUP) [en línea] Sucre,Bolivia: UNIVERSIDAD UNIÓN BOLIVARIANA. . Obtenido de <http://ingenieriadesoftware.mex.tl/images/18149/METODOLOGIAS%20AGILES.pdf>
- Rodriguez Donatien, A. (2011). Descripción de la Metodología de Desarrollo de Software Agile Unified Process (AUP) [en línea]. Recuperado el día 6 de Septiembre de 2013,. Obtenido de http://www.ecured.cu/index.php/Agile_Unified_Process
- Roger, P. (2005). Ingeniería del Software.6th.Ed.McGraw-Hill.
- Valerio, A. (4 de Marzo de 2005). Epidata Consulting. Obtenido de www.epidataconsulting.com

Anexo A

Especificación de Requerimientos según Norma IEEE 830

A. ESPECIFICACIÓN DE REQUERIMIENTOS IEEE 830

A.1. ESPECIFICACIÓN DE REQUERIMIENTOS MÓDULO ADMINISTRACIÓN

A.1.1. Gestión Módulo

Nombre: Gestión Módulo

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar) los módulos que la aplicación dispone.

Entrada: Código, nombre.

Proceso: El código será generado automáticamente, el nombre del módulo será validado y no se podrá ingresar otro con el mismo nombre, sino se ha llenado el campo nombre un mensaje será desplegado.

Se podrá actualizar el registro del módulo siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del módulo siempre y cuando no existan transacciones realizadas con esta.

El usuario de la gestión de los módulos será únicamente el administrador de la aplicación.

Salida: Módulo gestionado.

A.1.2. Gestión Opción

Nombre: Gestión Opción.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar) las opciones que la aplicación dispone, es decir los formularios y reportes creados.

Entrada: Código, Padre, Nombre Formulario, Posición, Url, Tipo, Estado, módulo.

Proceso: El código será generado automáticamente, el nombre de la opción será validado y no se podrá ingresar otro con el mismo nombre, se ingresara la posición dentro del menú, se ingresara la dirección url del formulario, el tipo si es formulario o reporte, el estado de la opción por defecto este estará activo y el módulo al cual pertenece esta opción.

Si no se ha llenado los campos requeridos un mensaje será desplegado.

Se podrá actualizar el registro de la opción siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro de la opción siempre y cuando no existan transacciones realizadas con esta.

El administrador es el único usuario que tendrá acceso a esta funcionalidad.

Salida: Opción gestionado.

A.1.3. Gestión Perfil

Nombre: Gestión Perfil.

Objetivo Gestionar (Crear, modificar, seleccionar, eliminar) los perfiles para los usuarios.

Entrada: Código, Nombre

Proceso: El código será generado automáticamente, el nombre del perfil será validado y no se podrá ingresar otro con el mismo nombre.

Si no se ha llenado los campos un mensaje será desplegado.

Se podrá actualizar el registro del perfil siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del perfil siempre y cuando no existan transacciones realizadas con esta.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Perfil gestionado.

A.1.4. Gestión Opción Perfil

Nombre: Gestión Opción Perfil.

Objetivo Gestionar (Crear, modificar, seleccionar, eliminar) las opciones asociadas a los perfiles.

Entrada: perfil, opción

Proceso: Serán asociadas las opciones a cada uno de los perfiles creados.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Opción asociado a un perfil.

A.1.5. Parámetros de seguridad

Nombre: Parámetros de seguridad.

Objetivo Parametrizar las opciones de seguridad para la aplicación.

Entrada: Tiempo de Vigencia contraseña, Tamaño Mínimo contraseña, Intentos Fallidos de conexión

Proceso: El usuario administrador es el encargado de configurar las variables de seguridad para la contraseña del usuario.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: parámetros configurados.

A.1.6. Gestión usuario

Nombre: Gestión usuario.

Objetivo Gestionar (Crear, modificar, seleccionar, eliminar) los usuarios de la aplicación.

Entrada: código de usuario, código de personal, código de estudiante, código de familiar, Username, contraseña, fecha de ingreso, fecha caducidad, perfil, estado, validez, intento Fallido Conexión.

Proceso: El usuario administrador o el usuario que el administrador haya designado como responsable de la creación de usuarios, es o son los encargados de gestionar (Crear, modificar, seleccionar, eliminar) los usuarios.

Los estudiantes, docentes, familiares, serán creados de manera automática como usuario de la aplicación.

Cuando se cree un usuario que no es estudiante, docente o familiar, se deberá ingresar el username, perfil para crear un usuario.

El sistema validara que no existan otros username con el mismo nombre, la contraseña será generada automáticamente y será la misma del username.

La fecha de ingreso será generada automáticamente.

El campo de validez fallida de conexión será actualizado cada vez que el usuario falle en la conexión hasta llegar a ser igual al configurado anteriormente.

Se podrá actualizar el registro del usuario siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del usuario siempre y cuando no existan transacciones realizadas con este.

Salida: Usuario gestionado.

A.1.7. Primer ingreso

Nombre: Primer ingreso.

Objetivo Asignar otra contraseña al usuario que ingreso por primera vez.

Entrada: Username, contraseña.

Proceso: El usuario digitara su username y contraseña, si la contraseña guardada en la base de datos es la misma del username y la contraseña digitada, entonces la aplicación mostrara otra pantalla en la cual el usuario deberá digitar la antigua contraseña y después digitar 2 veces la nueva contraseña.

El sistema validara la contraseña con los parámetros de configuración.

Si pasa las validaciones entonces la nueva contraseña será guardada en la base de datos.

Salida: Contraseña cambiado y encriptada.

A.1.8. Histórico usuario

Nombre: Histórico usuario.

Objetivo Llevar un registro histórico de las contraseñas generadas por los usuarios.

Entrada: código de usuario, fecha, contraseña encriptada.

Proceso: Cada vez que el usuario cambie su contraseña esta será guardada en una tabla histórica, con el fin de evitar que el usuario repita consecutivamente la misma contraseña, mejorando la seguridad de la aplicación.

Salida: Contraseña guardada históricamente.

A.2. ESPECIFICACIÓN DE REQUERIMIENTOS MÓDULO ACADÉMICO

A.2.1. Gestión Área Académica

Nombre: Gestionar Área Académica.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las áreas académicas.

Entrada: código área académica, nombre.

Proceso: El código será generado automáticamente, el nombre del área académica será validado y no se podrá ingresar otra con el mismo nombre.

Si no se ha llenado los campos un mensaje será desplegado.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Se podrá actualizar el registro del área académica siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del área académica siempre y cuando no existan transacciones realizadas con esta.

Salida: Área Académica creado.

A.2.2. Gestión Especialidad

Nombre: Gestión Especialidad.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las especialidades.

Entrada: código, nombre

Proceso: El código será generado automáticamente, el nombre de la especialidad será validado y no se podrá ingresar otra con el mismo nombre.

Si no se ha llenado los campos un mensaje será desplegado.

Se podrá actualizar el registro de la especialidad siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro de la especialidad siempre y cuando no existan transacciones realizadas con esta.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Especialidad gestionado.

A.2.3. Estructura Año

Nombre: Año Lectivo.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar) el año lectivo.

Entrada: código año lectivo, Fecha Inicio, Fecha Final, código Padre, Descripción, Nivel, Tipo, Nota Docente, vigente.

Proceso: El código será generado automáticamente, el nombre del año lectivo será validado y no se podrá ingresar otra con el mismo nombre.

Se definirá la fecha de inicio y la fecha de fin del año lectivo.

Se definirá el modo de evaluación y la distribución para cada uno de estos, por ejemplo: al año lectivo existirán 2 quimestres, cada uno de ellos con 3 parciales y cada parcial con varios aportes.

Cuando se configura el año lectivo este será colocado como vigente lo que indica que ya no se podrá hacer cambios en los años lectivos anteriores, estos solamente serán de lectura.

Si no se ha llenado los campos un mensaje será desplegado.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Año lectivo creado los otros años lectivos quedan como cerrados y solo de lectura.

Estructura Día

Nombre: Estructura Día.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) los días asignados al año lectivo vigente.

Entrada: código año lectivo, día, día asistencia.

Proceso: Una vez creado el año lectivo se asocia los días de la semana y si estos días son días laborables o no.

Este proceso será utilizado para la creación del horario de clases.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Días asignados al año lectivo vigente.

A.2.4. Estructura hora

Nombre: Estructura hora.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las horas asignados a la semana para el año lectivo vigente.

Entrada: código año lectivo, hora, hora inicio, hora fin, día asistencia.

Proceso: Una vez creado el año lectivo se asocia las horas con el tiempo que cada una tiene, es decir, hora inicio y hora fin y si estas horas son laborables o no.

Este proceso será utilizado para la creación del horario de clases.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Días asignados al año lectivo vigente.

A.2.5. Gestión Materia

Nombre: Gestión Materia.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las materias que los estudiantes recibirán.

Entrada: código materia, área académica, nombre materia, aplica grado, promedia, estado.

Proceso: El código será generado automáticamente, el nombre de la materia será validado y no se podrá ingresar otra con el mismo nombre.

Se seleccionara el área académica a la cual pertenece la materia, se seleccionara si esta materia aplica para grado y si promedia.

Se podrá actualizar el registro de la materia siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro de la materia siempre y cuando no existan transacciones realizadas con esta.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Días asignados al año lectivo vigente.

A.2.6. Gestión Nivel

Nombre: Gestión Nivel.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el nivel de educación (Bachillerato y Educación General Básica).

Entrada: código nivel, nombre nivel

Proceso: El código será generado automáticamente, el nombre del nivel será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del nivel siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del nivel siempre y cuando no existan transacciones realizadas con esta.

El administrador es el único usuario que tendrá acceso a esta funcionalidad

Salida: Nivel gestionado.

A.2.7. Gestión País

Nombre: Gestión País.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) del país.

Entrada: código país, nombre país

Proceso: El código será generado automáticamente, el nombre del país será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del país siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del país siempre y cuando no existan transacciones realizadas con este.

Salida: País gestionado.

A.2.8. Gestión Provincia

Nombre: Gestión Provincia.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) la provincia.

Entrada: código de la provincia, nombre de la provincia, código de país.

Proceso: El código será generado automáticamente, el nombre de la provincia será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro de la provincia siempre y cuando no exista otra con el mismo nombre.

Se asignará el país al cual pertenece esta provincia.

Se podrá eliminar el registro de la provincia siempre y cuando no existan transacciones realizadas con este.

Salida: provincia gestionada.

A.2.9. Gestión Cantón

Nombre: Gestión Cantón.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el cantón.

Entrada: código del cantón, nombre del cantón.

Proceso: El código será generado automáticamente, el nombre del cantón será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del cantón siempre y cuando no exista otra con el mismo nombre.

Se asignara la provincia al cual pertenece este cantón.

Se podrá eliminar el registro del cantón siempre y cuando no existan transacciones realizadas con este.

Salida: cantón gestionado.

A.2.10. Gestión Parroquia.

Nombre: Gestión Parroquia.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) la parroquia.

Entrada: código de la parroquia, nombre de la parroquia.

Proceso: El código será generado automáticamente, el nombre de la parroquia será validado y no se podrá ingresar otra con el mismo nombre.

Se asignara el cantón al cual pertenece esta parroquia.

Se podrá actualizar el registro de la parroquia siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro la parroquia siempre y cuando no existan transacciones realizadas con este.

Salida: Parroquia gestionado.

A.2.11. Gestión Barrio

Nombre: Gestión Barrio.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el barrio.

Entrada: código del barrio, nombre del barrio.

Proceso: El código será generado automáticamente, el nombre del barrio será validado y no se podrá ingresar otra con el mismo nombre.

Se asignara la parroquia a al cual pertenece este barrio.

Se podrá actualizar el registro del barrio siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro el barrio siempre y cuando no existan transacciones realizadas con este.

Salida: Barrio gestionado.

A.2.12. Gestión Cargo.

Nombre: Gestión Cargo.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el cargo, será utilizado en el módulo del DOBE para asignar el cargo que tienen los padres del estudiante en sus trabajos.

Entrada: código del barrio, nombre del barrio.

Proceso: El código será generado automáticamente, el nombre del cargo será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del cargo siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del cargo siempre y cuando no existan transacciones realizadas con este.

Salida: Cargo gestionado.

A.2.13. Estado Civil

Nombre: Estado Civil.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el estado civil.

Entrada: código del estado civil, nombre del estado civil.

Proceso: El código será generado automáticamente, el nombre del estado civil será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del estado civil siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del estado civil siempre y cuando no existan transacciones realizadas con este.

Salida: Estado civil gestionado.

A.2.14. Gestión Idioma

Nombre: Gestión Idioma.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) los avances del personal de los cursos de idiomas que sigan.

Entrada: código del idioma, nombre del idioma.

Proceso: El código será generado automáticamente, el nombre del idioma será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del idioma siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del idioma siempre y cuando no existan transacciones realizadas con este.

Salida: Idioma gestionado.

A.2.15. Gestión Institución

Nombre: Gestión Institución.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las instituciones donde los estudiantes anteriormente estudiaron, este proceso será utilizado en el módulo del DOBE.

Entrada: código de la institución, nombre de la institución.

Proceso: El código será generado automáticamente, el nombre de la institución será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro de la institución siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro la institución siempre y cuando no existan transacciones realizadas con este.

Salida: institución gestionado.

A.2.16. Gestión Instrucción

Nombre: Gestión Instrucción.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las instrucciones, es decir, el nivel de educación que tiene el familiar o el personal.

Entrada: código de instrucción, nombre de la instrucción.

Proceso: El código será generado automáticamente, el nombre de la instrucción será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro de la instrucción siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro de la instrucción siempre y cuando no existan transacciones realizadas con este.

Salida: instrucción gestionada.

A.2.17. Gestión Parentesco Familiar

Nombre: Gestión Parentesco Familiar.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el parentesco familiar.

Entrada: código de parentesco familiar, nombre del parentesco familiar.

Proceso: El código será generado automáticamente, el nombre del parentesco familiar será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del parentesco familiar siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del parentesco familiar siempre y cuando no existan transacciones realizadas con este.

Salida: parentesco familiar gestionado.

A.2.18. Gestión Raza

Nombre: Gestión Raza.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) la raza a la que pertenece cada estudiante. Sera utilizado por el módulo del DOBE en la Ficha Psicológica.

Entrada: código de raza, nombre de la raza.

Proceso: El código será generado automáticamente, el nombre de la raza será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro de la raza siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro de la raza siempre y cuando no existan transacciones realizadas con este.

Salida: raza gestionada.

A.2.19. Gestión Religión

Nombre: Gestión Religión.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) la religión a la que pertenece cada estudiante. Sera utilizado por el módulo del DOBE en la Ficha Psicológica.

Entrada: código de la religión, nombre de la religión.

Proceso: El código será generado automáticamente, el nombre de la religión será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro de la religión siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro de la religión siempre y cuando no existan transacciones realizadas con este.

Salida: religión gestionada

A.2.20. Gestión Tipo Domicilio

Nombre: Gestión Tipo Domicilio.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el tipo de domicilio en donde vive el estudiante.

Entrada: código del tipo de domicilio, nombre del tipo de domicilio.

Proceso: El código será generado automáticamente, el nombre del tipo de domicilio será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el registro del tipo de domicilio siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del tipo de domicilio siempre y cuando no existan transacciones realizadas con este.

Salida: tipo de domicilio gestionada.

A.2.21. Gestión Tipo Personal

Nombre: Gestión Tipo Personal.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el tipo de personal.

Entrada: código del tipo de personal, nombre del tipo de personal.

Proceso: El código será generado automáticamente, el nombre del tipo de personal será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el nombre del tipo de personal siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del tipo de personal siempre y cuando no existan transacciones realizadas con este.

Salida: tipo de personal gestionado.

A.2.22. Gestión Tipo Sangre

Nombre: Gestión Tipo Sangre.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el tipo de sangre.

Entrada: código del tipo de sangre, nombre del tipo de sangre.

Proceso: El código será generado automáticamente, el nombre del tipo de sangre será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el nombre del tipo de sangre siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del tipo de sangre siempre y cuando no existan transacciones realizadas con este.

Salida: tipo de sangre gestionado.

A.2.23. Gestión Tipo Teléfono

Nombre: Gestión Tipo Teléfono.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el tipo de teléfono.

Entrada: código del tipo de teléfono, nombre del tipo de teléfono.

Proceso: El código será generado automáticamente, el nombre del tipo de teléfono será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el nombre del tipo de teléfono siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del tipo de teléfono siempre y cuando no existan transacciones realizadas con este.

Salida: tipo de teléfono gestionado.

A.2.24. Gestión Estudiante

Nombre: Gestión Estudiante.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) estudiantes.

Entrada: código estudiante, tipo sangre, instituto, raza, religión, estado, Apellido Paterno, Apellido Materno, Nombres, Identificación, Sexo, Fecha Nacimiento, lugar nacimiento, Numero Calzado, Peso, Talla, color cabello, color ojos.

Proceso: El código será generado automáticamente, el nombre del estudiante será validado y se podrá ingresar otra con el mismo nombre, esto debido a que existen homónimos, es decir personas que tienen los mismos nombres y apellidos.

Además, se ingresaran otras variables como sexo, tipo sangre entre otras.

Se ingresara el número de identificación, este hace referencia a la cedula de identidad que debe ser única, no se podrá registrar dos estudiantes con el mismo número de identidad, para el caso de extranjeros se ingresara el número de pasaporte que no será validado.

Se podrá eliminar el registro del estudiante siempre y cuando no existan transacciones realizadas con este.

Salida: estudiante gestionado.

A.2.25. Gestión Personal

Nombre: Gestión Personal.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) personal.

Entrada: código personal, tipo sangre, raza ,tipo personal, religión, estado civil, estado, apellido paterno, apellido materno, nombres, identificación, sexo ,color piel, color ojos, color cabello, estatura, talla viste, calzado, fecha nacimiento, fecha ingreso, pertenece consejo, lugar nacimiento.

Proceso: El código será generado automáticamente, el nombre del personal será validado y se podrá ingresar otra con el mismo nombre, esto debido a que existen homónimos, es decir personas que tienen los mismos nombres y apellidos.

Además, se ingresaran otras variables como sexo, tipo sangre entre otras.

Se ingresara el número de identificación, este hace referencia a la cedula de identidad que debe ser única, no se podrá registrar dos estudiantes con el mismo número de identidad, para el caso de extranjeros se ingresara el número de pasaporte que no será validado.

Se podrá eliminar el registro del personal siempre y cuando no existan transacciones realizadas con este.

Salida: Personal gestionado.

A.2.26. Gestión Curso Paralelo

Nombre: Gestión Curso Paralelo.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) cursos y paralelos.

Entrada: Curso paralelo código, especialidad, año lectivo, dirigente de curso paralelo, código curso, curso paralelo nombre, número estudiantes.

Proceso: El código será generado automáticamente, el nombre del curso paralelo será validado y se no podrá ingresar otra con el mismo nombre para el año lectivo vigente.

Además, se ingresara la especialidad, el personal que será el dirigente de ese curso paralelo, la cantidad de estudiantes para ese curso-paralelo.

Se podrá eliminar el registro del curso paralelo siempre y cuando no existan transacciones realizadas con este.

Salida: curso – paralelo gestionado.

A.2.27. Estudiante Año Lectivo (Matriculación).

Nombre: Estudiante Año lectivo (Matriculación).

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) estudiantes en el año lectivo vigente.

Entrada: código curso paralelo, código estudiante, fecha Matricula, estado, número matricula.

Proceso: Una vez seleccionado el año lectivo vigente, el curso paralelo, podemos asignar el estudiante.

El sistema verificara si el estudiante puede matricularse en ese curso paralelo, si la respuesta es afirmativa, asignara el número de matrícula y la transacción quedara registrada con la fecha de la matrícula.

Además, se ingresara la especialidad, el personal que será el dirigente de ese curso paralelo, la cantidad de estudiantes para ese curso-paralelo.

Se podrá cambiar el curso paralelo asignado para el estudiante si no existen transacciones realizadas por ese estudiante en ese curso paralelo.

Se podrá eliminar la matricula siempre y cuando no existan transacciones realizadas con este.

Salida: matricula asignada.

A.2.28. Asignación de Materia – Docente, Curso - Paralelo

Nombre: Asignación Materia – Docente, Curso – Paralelo.

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) materias asignadas a un profesor en un curso - paralelo.

Entrada: código materia curso - paralelo, código materia, código curso - paralelo, código personal, materia grado.

Proceso: Una vez seleccionado el año lectivo vigente, el curso paralelo, podemos asignar la materia y el docente que será el encargado de dictar clases en ese curso - paralelo.

El código de materia curso - paralelo será generado automáticamente.

Además, se ingresara si la materia es parte de las materias de grado.

El sistema verificara la integridad de los datos ya que no se puede ingresar un curso – paralelo para el mismo docente y materia dos veces.

Se podrá cambiar el curso paralelo asignado para la materia si no existen transacciones realizadas por esa materia en ese curso paralelo.

Se podrá eliminar la asociación entre materia-docente, curso-paralelo siempre y cuando no existan transacciones realizadas con este.

Salida: Materias, Docentes asignados a cursos – Paralelos.

A.2.29. Asignación del Horario de Clases

Nombre: Asignación del Horario de Clases

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) El horario de clases.

Entrada: código horario de clases, año lectivo, hora, día, código materia, código curso paralelo, código personal.

Proceso: Una vez seleccionado el año lectivo vigente, el curso paralelo, podemos asignar la materia y el docente para un día y una hora específica.

El horario de clase nos indica que un docente dictara una materia en un curso paralelo en un día y hora específica en la semana. Y este no debe cruzarse con ningún otro docente en otra hora, día, materia y curso – paralelo.

El sistema verificara la integridad de los datos ya que no se puede ingresar un curso – paralelo para el mismo docente, materia, día y hora dos veces.

Se podrá eliminar la asociación el horario de clases siempre y cuando no existan transacciones realizadas con este.

Salida: Horario de clases para el año lectivo vigente.

A.2.30. Registro de Notas

Nombre: Registros de Notas

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las notas asignadas a los estudiantes.

Entrada: Año lectivo, código estudiante, código materia, código curso paralelo, Fecha, Nota.

Proceso: El docente ingresara a la aplicación y seleccionara el curso paralelo, dependiendo de la configuración del año lectivo tendrá varias opciones como quimestres, parciales, aportes, exámenes. Registrará la o las notas de los estudiantes matriculados en ese curso – paralelo.

Los cálculos se realizaran de acuerdo a las configuraciones del año lectivo.

El sistema guardara la información, el docente podrá guardar las notas de otros alumnos de otros cursos y paralelos.

El docente podrá ingresar y corregir las notas guardadas anteriormente hasta que el año lectivo se encuentre vigente.

El docente podrá solo visualizar información de años lectivos anteriores.

Salida: Registro de notas.

A.3. ESPECIFICACIÓN DE REQUERIMIENTOS MÓDULO DOBE

A.3.1. Gestión Ficha Psicológica

Nombre: Gestión Ficha Psicológica

Objetivo: Gestionar (Crear, modificar, seleccionar, buscar) la ficha psicológica de los estudiantes.

Entrada: código estudiante, código de la institución anterior, tiene compañía al llegar a casa, cuantas horas pasa solo, tipo de relación con la madre, tipo

de relación con el padre, tipo de relación con los hermanos, tipo de hogar, tipo de embarazo de la madre, enfermedad, en caso de afirmativo que o cuales enfermedades, edad gestacional del estudiante, tipo de parto, problemas de salud antes de cumplir el año de vida, tipo de alimentación, si gateo o no, edad que camino, edad hablo claramente, estuvo en guardería, problemas emocionales , problemas de conducta, problemas de lenguaje, deficiencia visual, deficiencia auditiva, problemas salud, impedimentos físicos, cantidad de amigos que tiene, actividades extra curriculares, horas dedicadas al estudio, condiciones de vivienda.

Proceso: El usuario, ingresara los datos anteriormente indicados para llenar correctamente la ficha, el ingreso de los datos puede ser de diferentes tipos: cajas de texto para llenar información, listas desplegables para seleccionar una opción, lista de chequeos para seleccionar más de una opción.

El usuario podrá ingresar y modificar la información de la ficha de psicología.

El usuario no podrá eliminar la ficha de psicología.

Salida: Ficha Psicológica gestionada.

A.3.2. Gestión tipo caso

Nombre: Gestión tipo caso

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) los tipos de casos.

Entrada: código del tipo de caso, nombre del tipo de caso.

Proceso: El código será generado automáticamente, el nombre del tipo de caso será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el nombre del tipo de caso siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del tipo de caso siempre y cuando no existan transacciones realizadas con este.

Salida: tipo de caso gestionado.

A.3.3. Gestión Caso

Nombre: Gestión Caso

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) los casos asignados por psicología a los estudiantes.

Entrada: código del caso, fecha de inicio del caso, fecha de fin del caso, descripción del caso, año lectivo, código del estudiante, código del psicólogo, estado del caso, tipo de caso (normal, especial).

Proceso: El código del caso será generado automáticamente, el psicólogo deberá buscar y seleccionar al estudiante, la fecha de inicio será creada al momento de guardar el caso, por defecto el estado del caso quedar como iniciado, el psicólogo deberá ingresar un detalle del caso, además, deberá seleccionar el tipo de caso.

Una vez creado el caso, el usuario podrá ingresar las actividades a realizarse para dicho caso, este proceso será detallado más abajo.

El usuario ingresara al caso y podrá realizar cambios, si y solo si, no existan ninguna actividad para este caso, caso contrario solo podrá cambiar de estado.

Salida: Caso gestionado.

A.3.4. Gestión Caso Detalle (actividades)

Nombre: Gestión Caso Detalle

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) el detalle de los casos (actividades) asignados por psicología a los estudiantes.

Entrada: código del caso detalle (actividad), fecha de la actividad, descripción de la actividad, terapia a realizar.

Proceso: Una vez creado el caso, el usuario podrá ingresar las actividades a realizarse para dicho caso, el usuario ingresara la fecha de la actividad, el detalle de la actividad, si la actividad necesita algún tipo de terapia se ingresara la terapia.

Las actividades pueden ser modificadas y eliminadas si el usuario lo considera necesario.

Los casos ya no podrán ser modificados cuando el estado cambie a cerrado.

Los casos podrán permanecer abiertos incluso cuando el año lectivo vigente haya terminado.

Salida: Gestión detalle de Casos (actividades).

A.3.5. Gestión Terapias

Nombre: Gestión Terapias

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las terapias de las actividades creadas.

Entrada: código de la terapia, descripción de la terapia.

Proceso: Dependiendo de la gravedad de caso, los casos pueden o no, tener terapias.

Las terapias serán asignadas a las actividades una vez creadas estas.

Las terapias son optativas, el psicólogo definirá si es necesario aplicarla o no.

Salida: terapias gestionadas.

A.3.6. Gestión Visitas

Nombre: Gestión Visitas

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) las visitas realizadas por los psicólogos a los domicilios.

Entrada: código de la visita, fecha de creación, fecha de visita, hora de visita, observaciones, docente asignado, estudiante a visitar, horario de clase, familia que fue entrevistada, motivo.

Proceso: El psicólogo ingresara los datos del estudiante, el docente encargado de la visita, las observaciones de la visita, la fecha y hora de la visita, quien estuvo en el hogar cuando se realizó esta, el motivo de la visita.

Salida: visitas gestionadas.

A.3.7. Gestión motivo

Nombre: Gestión motivo

Objetivo: Gestionar (Crear, modificar, seleccionar, eliminar, buscar) los motivos de las visitas.

Entrada: código del motivo, nombre del motivo.

Proceso: El código será generado automáticamente, el nombre del motivo será validado y no se podrá ingresar otra con el mismo nombre.

Se podrá actualizar el nombre del motivo siempre y cuando no exista otra con el mismo nombre.

Se podrá eliminar el registro del motivo siempre y cuando no existan transacciones realizadas con este.

Salida: motivo gestionado.

Anexo B

Especificación de Casos de Uso y Diagramas

B. Especificación de Casos de Uso

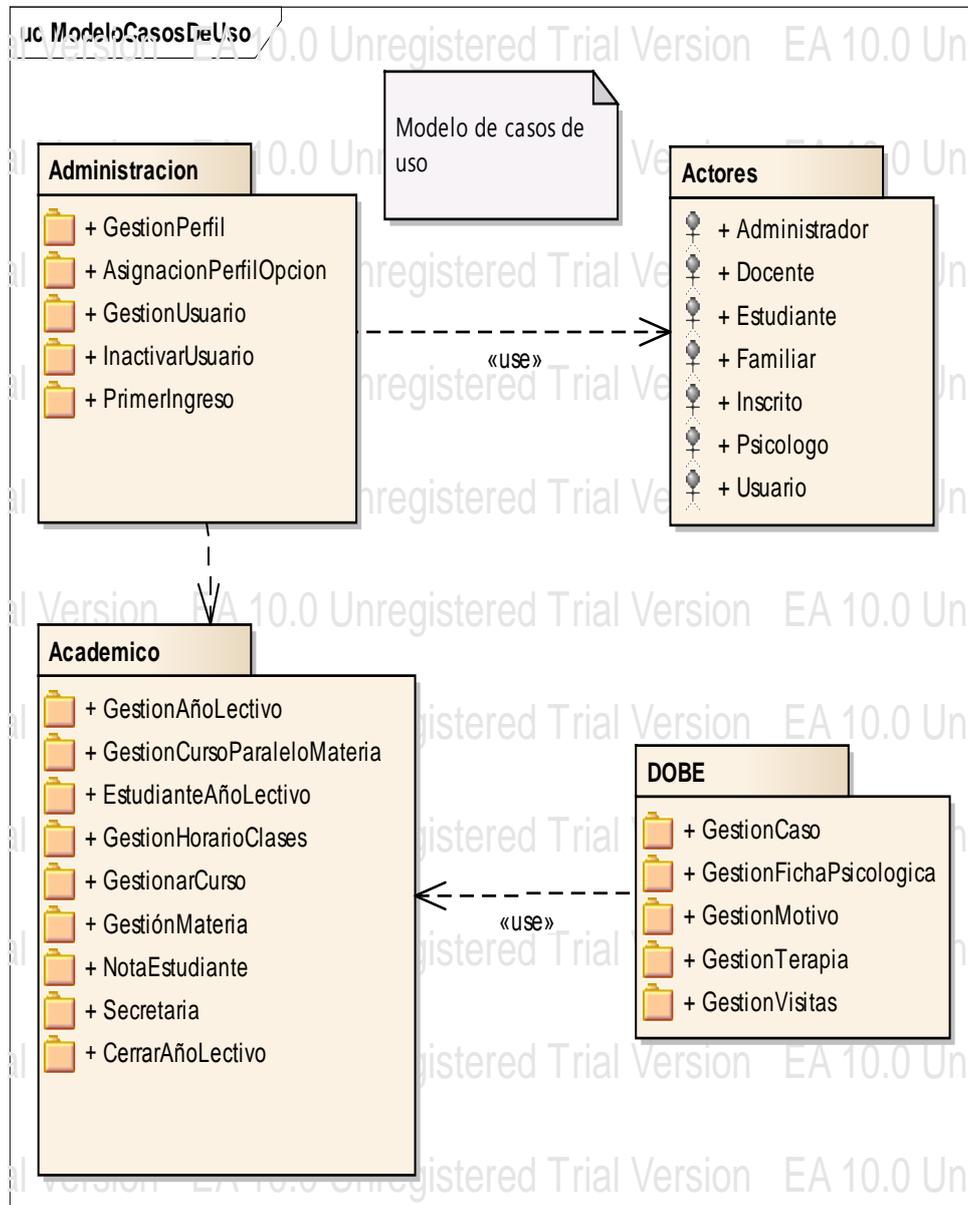


Figura B. 1 Paquetes Módulos de la Aplicación

B.1. Especificación de Casos de Uso Modulo de Administración

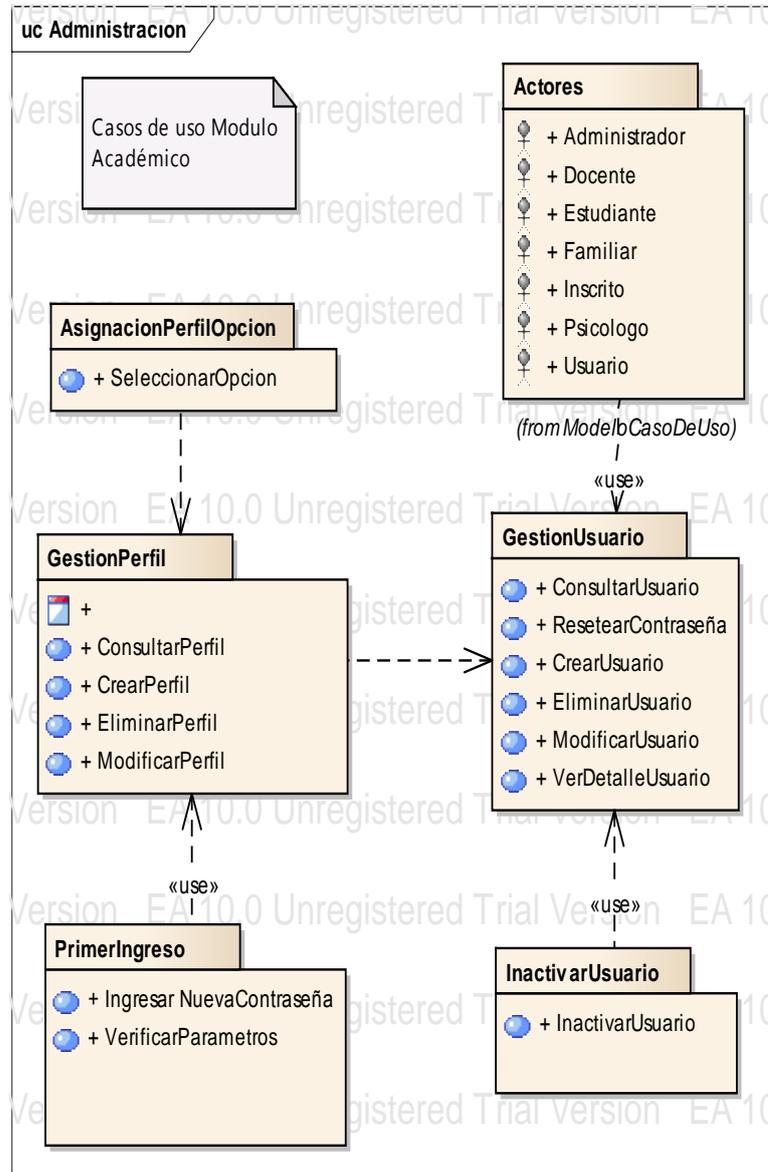


Figura B. 2 Paquetes Módulo Administración

B.1.1. Caso de Uso CrearPerfil

Módulo de Administración	
A.1.3 Gestión Perfil	
Caso de Uso	CrearPerfil
Actor	Administrador
Tipo	
Descripción	Se realiza la creación del Perfil.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un perfil de usuario
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del perfil que desea crear.	Si el nombre del perfil ya existe, un mensaje de error se desplegará indicando que ese perfil ya se encuentra creado. Si el nombre del perfil no existe, entonces el perfil se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los perfiles. No se puede crear más de un perfil con el mismo nombre.	

B.1.2. Caso de Uso ActualizarPerfil

Módulo de Administración	
A.1.3 Gestión Perfil	
Caso de Uso	ActualizarPerfil
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización del Perfil.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un perfil de usuario
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del perfil seleccionado.	Si el nombre del perfil ya existe, un mensaje de error se desplegará indicando que ese perfil ya se encuentra creado.

Continúa →

	Si el nombre del perfil no existe, entonces el perfil se actualizara en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los perfiles. No se puede actualizar más de un perfil con el mismo nombre.	

B.1.3. Caso de Uso ConsultarPerfil

Módulo de Administración	
A.1.3 Gestión Perfil	
Caso de Uso	ConsultarPerfil
Actor	Administrador
Tipo	
Descripción	Se realiza la consulta del Perfil.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un perfil de usuario.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del perfil.	Si el nombre o código del perfil existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del perfil no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los perfiles.	

B.1.4. Caso de Uso EliminarPerfil

Módulo de Administración	
A.1.3 Gestión Perfil	
Caso de Uso	EliminarPerfil
Actor	Administrador
Tipo	
Descripción	Se realiza la eliminacion del Perfil.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un perfil de usuario.



Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un perfil.	<p>Continúa</p> <p>Si el código del perfil existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese perfil tiene transacciones.</p> <p>Si el código del perfil existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el perfil.</p>
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los perfiles. No se podrá eliminar un perfil si tiene transacciones asociadas.	

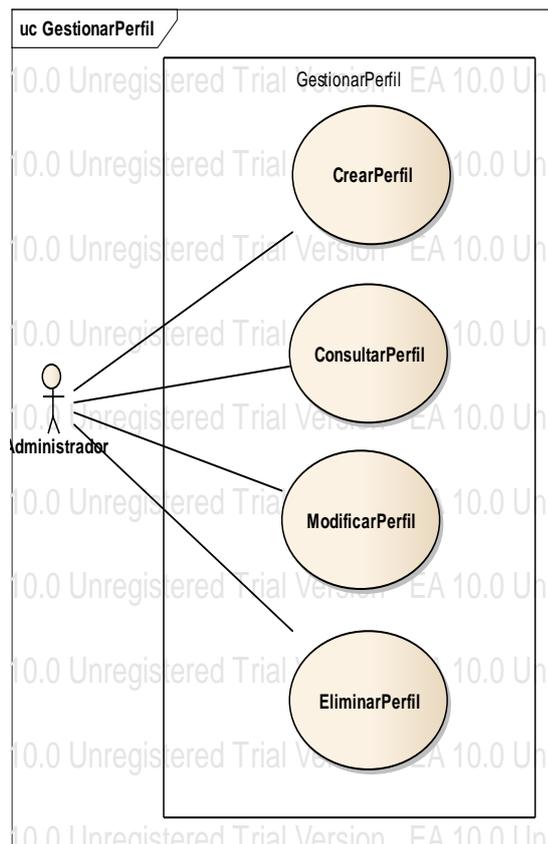


Figura B. 3 Caso de Uso Gestionar Perfil

B.1.5. Caso de Uso AsignarPerfilOpcion

Módulo de Administración	
A.1.4 Gestión Opción Perfil	
Caso de Uso	AsignarPerfilOpcion
Actor	Administrador
Tipo	
Descripción	Asignar al usuario los formularios y reportes.
Precondición	El Administrador ha ingresado al sistema
Post Condición	Ninguna.
Acción del Actor	Respuesta del Sistema
El administrador, debe asociar los perfiles a las opciones. El usuario del módulo tendrá acceso a los formularios a los cuales fue asignado mediante el perfil y las opciones. El administrador podrá cambiar de perfil a un usuario cuando este lo considere necesario.	El sistema mostrara el formulario de opciones, es decir, los formularios y reportes, que dispone la aplicación.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los perfiles y las opciones.	

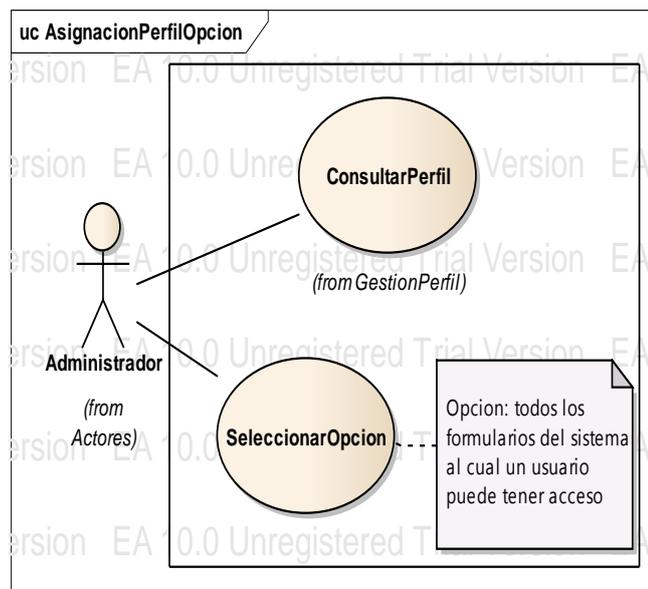


Figura B. 4 Caso de Uso Asignar Perfil Opción

B.1.6. Caso de Uso CrearUsuario

Módulo de Administración	
A.1.6 Gestión usuario	
Caso de Uso	CrearUsuario
Actor	Administrador
Tipo	
Descripción	Se realiza la creación de usuario
Precondición	El Administrador ha ingresado al sistema, gestión de perfiles y opciones.
Post Condición	Ninguna.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del usuario que desea crear. Al momento de crear usuarios el administrador colocara el username, el nombre del usuario, el estado (activo, inactivo), y el perfil al cual está asociado el usuario. Cuando el usuario es creado la contraseña asignada será la misma de su username.	Si el nombre del usuario ya existe, un mensaje de error se desplegará indicando que ese usuario ya se encuentra creado. Si el nombre del usuario no existe, entonces el usuario se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El usuario debe registrarse solo una vez. El Administrador es el responsable de la creación de los usuarios. El Administrador no podrá asignar la contraseña a los usuarios.	

B.1.7. Caso de Uso ActualizarUsuario

Módulo de Administración	
A.1.6 Gestión usuario	
Caso de Uso	ActualizarUsuario
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización del usuario.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un usuario del sistema.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del usuario	Si el nombre del usuario ya existe, un mensaje de error se desplegará. Continúa →

seleccionado. El administrador puede cambiar el perfil asociado al usuario. El administrador puede cambiar el estado del usuario	indicando que ese usuario ya se encuentra creado. Si el nombre del usuario no existe, entonces el usuario se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los usuarios. No se puede actualizar más de un usuario con el mismo nombre.	

B.1.8. Caso de Uso ConsultarUsuario

Módulo de Administración	
A.1.6 Gestión usuario	
Caso de Uso	ConsultarUsuario
Actor	Administrador
Tipo	
Descripción	Se realiza consulta de usuario.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un usuario del sistema.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del usuario.	Si el nombre o código del usuario existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del usuario no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los usuarios.	

B.1.9. Caso de Uso EliminarUsuario

Módulo de Administración

A.1.6 Gestión usuario	
Caso de Uso	EliminarUsuario
Actor	Administrador
Tipo	
Descripción	Se realiza la eliminación de usuario
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un usuario del sistema.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un usuario.	Si el código del usuario existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese usuario tiene transacciones. Si el código del usuario existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el usuario.
Observaciones	
El Administrador es el responsable de la gestión de los usuarios. No se podrá eliminar un usuario si tiene transacciones asociadas.	

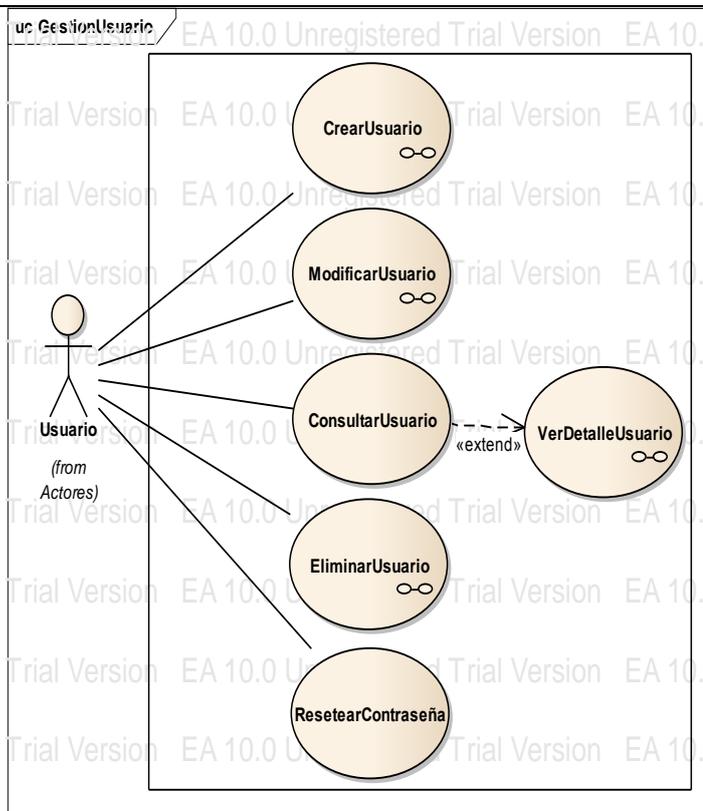


Figura B. 5 Caso de Uso Gestionar Usuario

B.1.10. Caso de Uso ResetearContraseñaUsuario

Módulo de Administración	
A.1.6 Gestión usuario	
Caso de Uso	ResetearContraseñaUsuario
Actor	Administrador
Tipo	
Descripción	Se resetea la contraseña de usuario.
Precondición	El Administrador ha ingresado al sistema. Usuario creado.
Post Condición	Se ha reseteado la contraseña de un usuario.
Acción del Actor	Respuesta del Sistema
El administrador del sistema reseteará la contraseña de un usuario.	Resetear la contraseña consiste en colocar el nombre del username ejemplo (oaltamirano) en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los usuarios.	

B.1.11. Caso de Uso IngresarNuevaContraseña

Módulo de Administración	
A.1.7 Primer Ingreso	
Caso de Uso	IngresarNuevaContraseña
Actor	Administrador
Tipo	
Descripción	Se realiza cambio de contraseña
Precondición	El Administrador ha ingresado al sistema. Usuario creado.
Post Condición	Se ha cambiado la contraseña de un usuario.
Acción del Actor	Respuesta del Sistema
El usuario en la pantalla inicial de acceso, digitara su username y la contraseña.	Si el username y la contraseña son ingresados correctamente, se desplegara un popup para que ingrese nuevamente la contraseña que fue asignado al momento de su creación, además, digitara su nueva contraseña y la confirmara. Si la contraseña es incorrecta se desplegara un mensaje de error de contraseña incorrecta. Si la contraseña es nueva y no coincide con la confirmación del nueva contraseña se desplegara un mensaje de error de no coincidencia entre contraseñas.

	Si la nueva contraseña y la confirmación coinciden este quedara grabado en la base de datos, y el usuario tendrá acceso a la aplicación y a los formularios asignados a su perfil.
Excepciones	
Ninguna	
Observaciones	
La contraseña al momento del reseteo es la misma del username. El Administrador es el responsable de la gestión de los usuarios. El Administrador no podrá asignar la contraseña a los usuarios	

B.1.12. Caso de Uso VerificarParametros

Módulo de Administración	
A.1.7 Primer Ingreso	
Caso de Uso	VerificarParametros
Actor	Administrador
Tipo	
Descripción	Se realiza verificación de parámetros de contraseña
Precondición	El Administrador ha ingresado al sistema. Usuario creado.
Post Condición	Se ha verificado los parámetros de acceso.
Acción del Actor	Respuesta del Sistema
Cuando el usuario ingrese su nombre de usuario y contraseña, se verificara el cumplimiento de las configuraciones del módulo de Administración. Se realizara la verificación de longitud de contraseña, número de intentos fallidos, estado del usuario.	Si la contraseña no cumple con las configuraciones se mostrara un mensaje de error. Si la contraseña cumple con las configuraciones, se tendrá acceso a los módulos asociados al perfil asignado.
Excepciones	
Ninguna	
Observaciones	
La contraseña al momento del reseteo es la misma del username. El Administrador es el responsable de la gestión de los usuarios. El Administrador no podrá asignar la contraseña a los usuarios.	

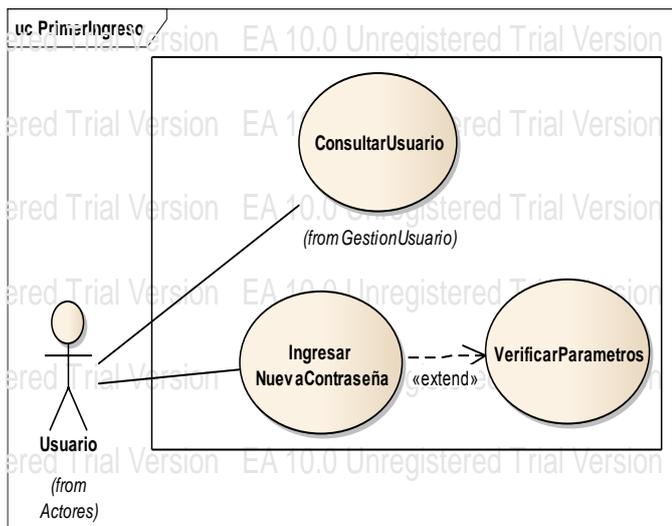


Figura B. 6 Caso de Uso Primer Ingreso

B.1.13. Caso de Uso InactivarUsuario

Módulo de Administración	
Caso de Uso	InactivarUsuario
Actor	Administrador
Tipo	
Descripción	Inactivar la cuenta del usuario
Precondición	El Administrador ha ingresado al sistema. El cliente debe haber notificado vía mail al administrador del sistema, que su cuenta de usuario sea dada de baja.
Post Condición	Se ha inactivado el usuario.
Acción del Actor	Respuesta del Sistema
El usuario seleccionara el usuario que desee inactivar.	El sistema mostrara un mensaje de aprobación del usuario inactivo.
Excepciones	
Cuando la cuenta de usuario todavía se esté usando normalmente por parte del cliente.	
Observaciones	
Ningún usuario podrá ser eliminado físicamente de la base de Datos. El usuario pasará a un estado de inactivo sólo si él mismo lo desea notificando de ésta necesidad al Administrador del Sistema.	

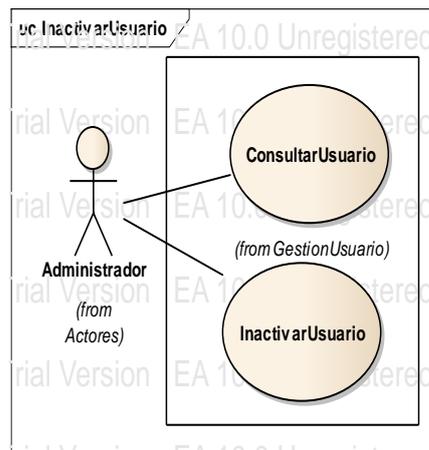


Figura B. 7 Caso de Uso Inactivar Usuario

B.2. Especificación de Casos de Uso del Modelo Académico

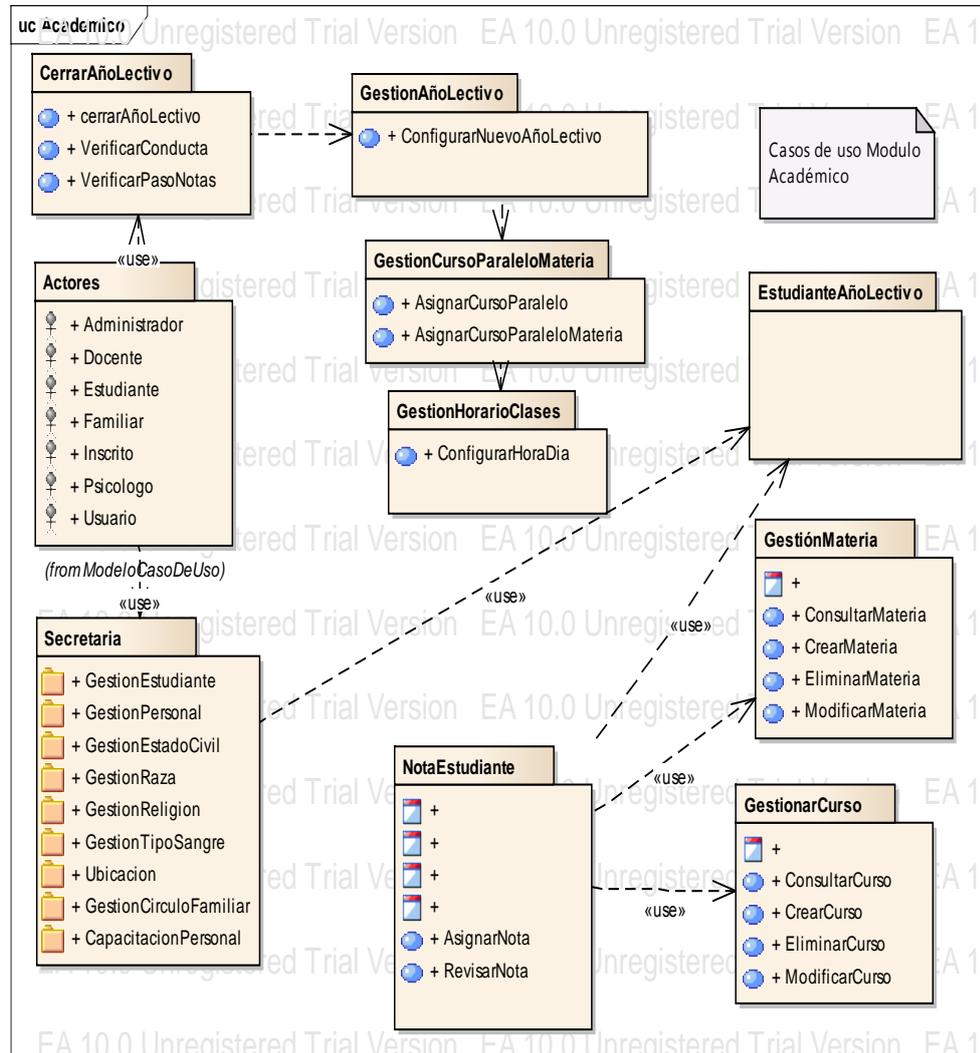


Figura B. 8 Paquetes Módulo Académico

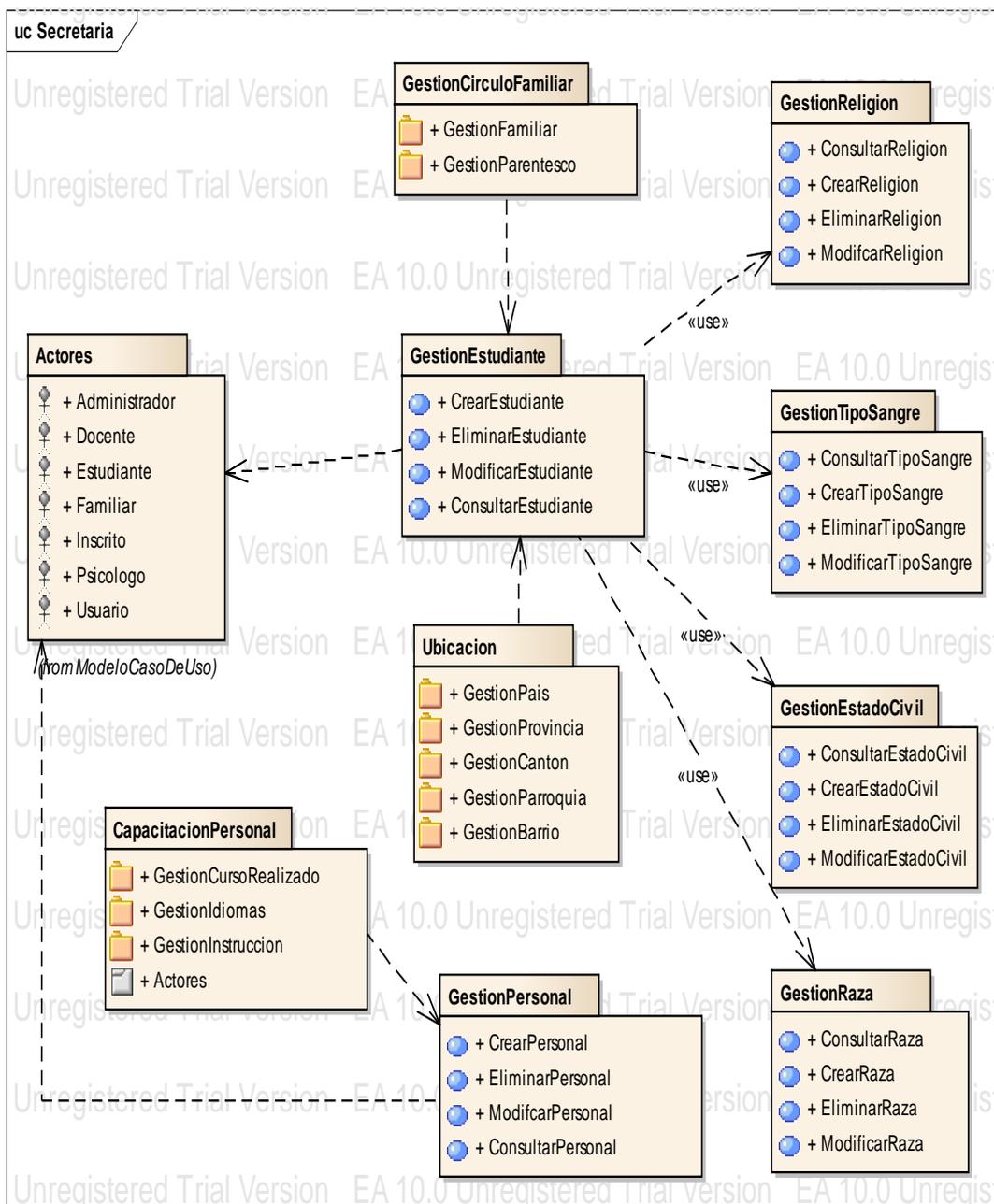


Figura B. 9 Paquetes Sub Módulo Secretaria

B.2.1. Casos de Uso CrearEstudiante

Sub Modulo Secretaria	
A.2.24 Gestión Estudiante	
Caso de Uso	CrearEstudiante
Actor	Secretaria
Tipo	
Descripción	Se creara un nuevo estudiante
Precondición	El usuario ha ingresado al sistema.
Post Condición	Se ha creado un estudiante.

Continúa →

Acción del Actor	Respuesta del Sistema
<p>Los usuarios (secretarias) ingresaran la información de los estudiantes.</p> <p>El usuario ingresara la cédula del estudiante. Si el estudiante no dispone de cédula, el usuario puede omitir la validación.</p> <p>El usuario ingresara la fecha de nacimiento. El usuario debe seleccionar el lugar de nacimiento del estudiantes, en el caso de no encontrar se debe dirigir al proceso de ubicación. El usuario debe seleccionar el lugar de domicilio del estudiante, en el caso de no encontrar se debe dirigir al proceso de ubicación.</p>	<p>El módulo validara la información ingresada y mostrara mensajes en caso de que la información sea errónea. El módulo validara la cédula.</p> <p>El módulo validara que se encuentre dentro de los parámetros permitidos, es decir, no permitirá edades negativas.</p> <p>El módulo validara la información y guardara. El módulo creara un código para cada estudiante. No se podrá guardar dos estudiantes con el mismo número de documento (identificación).</p>
Excepciones	
Ninguna	
Observaciones	
<p>El usuario es el responsable de la gestión de los estudiantes. No se puede crear más de un estudiante con el mismo número de identificación (cedula).</p>	

B.2.2. Caso de Uso ActualizarEstudiante

Sub Modulo Secretaria	
Gestión Estudiante	
Caso de Uso	ActualizarEstudiante
Actor	Secretaria
Tipo	
Descripción	Se realiza actualización de información estudiante.
Precondición	El usuario ha ingresado al sistema.
Post Condición	Se ha actualizado un estudiante.
Acción del Actor	Respuesta del Sistema
El usuario del sistema actualizará la información del estudiante seleccionado.	<p>Si la información del estudiante es incorrecta, un mensaje de error se desplegará indicando el campo donde se encuentra el error. Si la información del estudiante es correcta, entonces el estudiante se actualizará en el sistema.</p>

Continúa →

Excepciones Ninguna
Observaciones
La Secretaria es el responsable de la gestión de los estudiantes. No puede existir más de un estudiante con el mismo número de identificación

B.2.3. Caso de Uso ConsultarEstudiante

Sub Modulo Secretaria	
A.2.24 Gestión Estudiante	
Caso de Uso	ConsultarEstudiante
Actor	Secretaria
Tipo	
Descripción	Se realiza consulta de estudiante.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un estudiante.
Acción del Actor	Respuesta del Sistema
El usuario (Secretaria) del sistema consultará por varios criterios de búsqueda: nombre, apellido, número de identificación, código del estudiante.	El sistema desplegara en una grilla los resultados de la operación. Si bajo estos criterios no se desplegaré datos es posible que no existan estudiantes con los criterios de búsqueda ingresados.
Excepciones	
Ninguna	
Observaciones	
La Secretaria es el responsable de la gestión de los estudiantes.	

B.2.4. Caso de Uso EliminarEstudiante

Sub Modulo Secretaria	
A.2.24 Gestión Estudiante	
Caso de Uso	EliminarEstudiante
Actor	Secretaria
Tipo	Se realiza eliminación estudiante.
Descripción	
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un estudiante.
Acción del Actor	Respuesta del Sistema
El usuario (Secretaria) del sistema eliminará un estudiante.	Si el código del estudiante existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese estudiante tiene transacciones. Si el código del estudiante existe y este no tiene transacciones

	realizadas, un mensaje se desplegará indicando que se eliminó el estudiante.
Excepciones	
Ninguna	
Observaciones	
La Secretaria es el responsable de la gestión de los estudiantes. No se podrá eliminar un estudiante si tiene transacciones asociadas.	

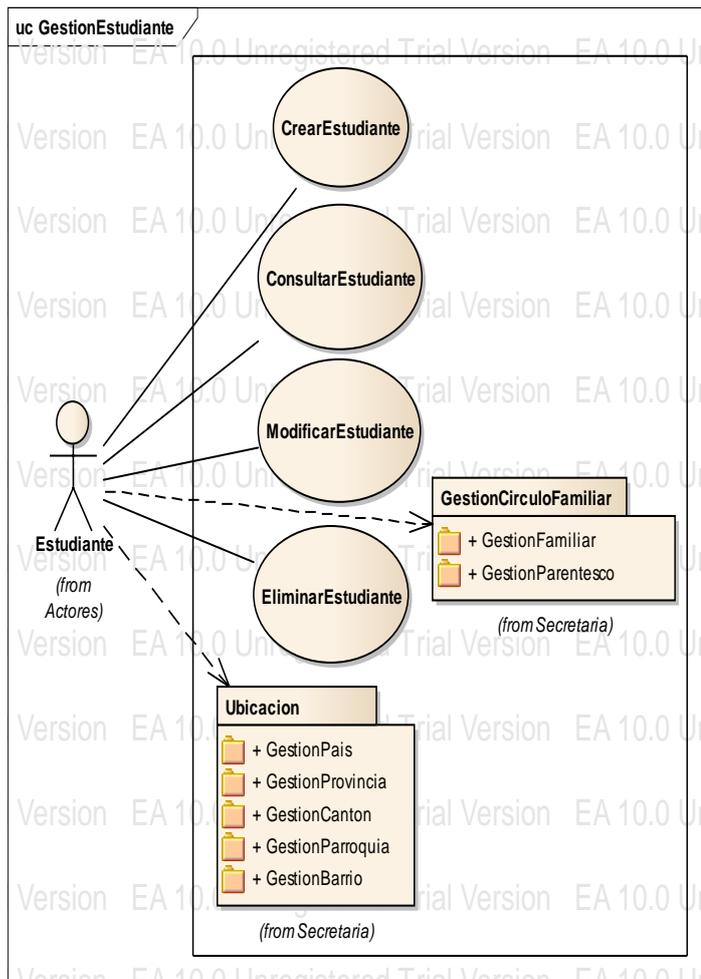


Figura B. 10 Caso de Uso Gestionar Estudiante

B.2.5. Caso de Uso CrearPersonal

Sub Modulo Secretaria	
A.2.25 Gestión Personal	
Caso de Uso	CrearPersonal
Actor	Secretaria
Tipo	
Descripción	Crear un personal: docente, psicólogo, administrativo
Precondición	La secretaria ha ingresado al sistema.
	Continúa →

Post Condición	Se ha creado un personal.	
Acción del Actor	Respuesta del Sistema	
Los usuarios (secretarias) ingresaran la información del personal. El usuario ingresara la fecha de nacimiento. El usuario debe seleccionar el lugar de nacimiento del personal, en el caso de no encontrar se debe dirigir al proceso de ubicación. El usuario debe seleccionar el lugar de domicilio del personal, en el caso de no encontrar se debe dirigir al proceso de ubicación.	El módulo validara la información ingresada y mostrara mensajes en caso de que la información sea errónea. El usuario ingresara la cédula del personal. El módulo validara la cédula. Si el personal no dispone de cédula o este es un pasaporte, el usuario puede omitir la validación. El módulo validara que se encuentre dentro de los parámetros permitidos, es decir, no permitirá edades negativas. El módulo validara la información y guardara. El módulo creara un código para cada personal.	
Excepciones		
Ninguna		
Observaciones		
Secretaria es el responsable de la gestión del personal. No se puede crear más de un personal con el mismo número de identificación (cedula).		

B.2.6. Caso de Uso ActualizarPersonal

Sub Modulo Secretaria		
A.2.25 Gestión Personal		
Caso de Uso	ActualizarPersonal	
Actor	Administrador	
Tipo		
Descripción	Secretaria realiza actualización de información personal.	
Precondición	La Secretaria ha ingresado al sistema.	
Post Condición	Se ha actualizado un personal.	
Acción del Actor	Respuesta del Sistema	
El usuario (Secretaria) actualizará la información del personal seleccionado. en el sistema.	Si la información del personal es incorrecta, un mensaje de error se desplegará indicando el campo donde se encuentra el error.	

	Si la información del personal es correcta, entonces el registro se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
La secretaria es el responsable de la gestión del personal. No puede existir más de un personal con el mismo número de identificación	

B.2.7. Caso de Uso ConsultarPersonal

Sub Modulo Secretaria	
A.2.25 Gestión Personal	
Caso de Uso	ConsultarPersonal
Actor	Secretaria
Tipo	
Descripción	Se realiza la consulta de personal.
Precondición	La Secretaria ha ingresado al sistema.
Post Condición	Se ha consultado un personal.
Acción del Actor	Respuesta del Sistema
El usuario (Secretaria) consultará por varios criterios un personal en el sistema.	Si el nombre, apellido, número de identificación, código del personal existe, se desplegara en una grilla los resultados de la operación. Si el nombre, apellido, número de identificación, código del personal no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
Secretaria es el responsable de la gestión del personal.	

B.2.8. Caso de Uso EliminarPersonal

Sub Modulo Secretaria	
A.2.25 Gestión Personal	
Caso de Uso	EliminarPersonal
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de personal.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un personal.
Acción del Actor	Respuesta del Sistema

<p>El usuario Secretaria eliminará un personal del sistema.</p>	<p>Si el código del personal existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese personal tiene transacciones. Si el código del personal existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el personal.</p>
<p>Excepciones</p>	
<p>Ninguna</p>	
<p>Observaciones</p>	
<p>La Secretaria es el responsable de la gestión del personal. No se podrá eliminar un personal si tiene transacciones asociadas.</p>	

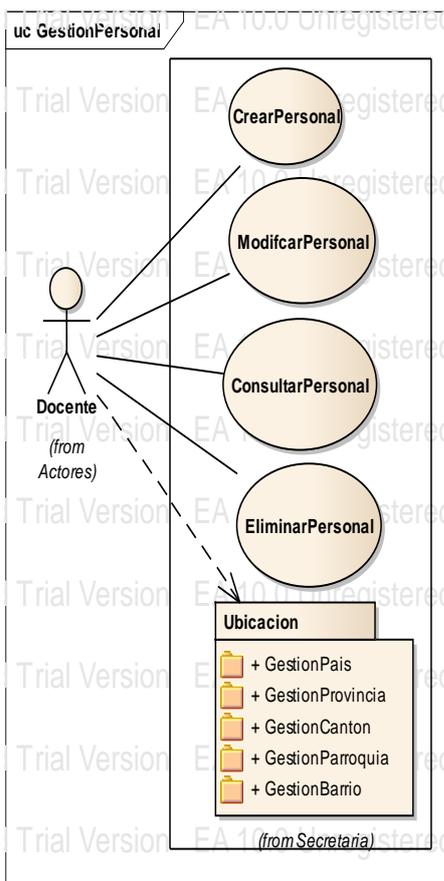


Figura B. 11 Caso de Uso Gestionar Personal

B.2.9. Caso de Uso CrearEstadoCivil

Sub Modulo Secretaria	
A.2.13 Gestión Estado Civil	
Caso de Uso	CrearEstadoCivil
Actor	Administrador
Tipo	
Descripción	
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un estado civil.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema ingresará el nombre del estado civil que desea crear.	Si el nombre del estado civil ya existe, un mensaje de error se desplegará indicando que ese estado civil ya se encuentra creado. Si el nombre del estado civil no existe, entonces el estado civil se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los estados civiles. No se puede crear más de un estado civil con el mismo nombre.	

B.2.10. Caso de Uso ActualizarEstadoCivil

Sub Modulo Secretaria	
A.2.13 Gestión Estado Civil	
Caso de Uso	ActualizarEstadoCivil
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización estado civil.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un estado civil.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema actualizará el nombre del estado civil seleccionado.	Si el nombre del estado civil ya existe, un mensaje de error se desplegará indicando que ese estado civil ya se encuentra creado. Si el nombre del estado civil no existe, entonces el estado civil se actualizará en el sistema.
Excepciones Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los estados civiles. No se puede actualizar más de un estado civil con el mismo nombre.	

B.2.11. Caso de Uso ConsultarEstadoCivil

Sub Modulo Secretaria	
A.2.13 Gestión Estado Civil	
Caso de Uso	ConsultarEstadoCivil
Actor	Secretaria
Tipo	
Descripción	Se realiza consulta de estado civil.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un estado civil.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema consultará el nombre o código del estado civil.	Si el nombre o código del estado civil existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del estado civil no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los estados civiles.	

B.2.12. Caso de Uso EliminarEstadoCivil

Sub Modulo Secretaria	
A.2.13 Gestión Estado Civil	
Caso de Uso	EliminarEstadoCivil
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación estado civil
Precondición	El Administrador ha ingresado al sistema
Post Condición	Se ha eliminado un estado civil.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema eliminará un estado civil.	Si el código del estado civil existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese registro tiene transacciones. Si el código del estado civil existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	Ninguna
Observaciones	
El Administrador es el responsable de la gestión de los estados civiles. No se podrá eliminar un estado civil si tiene transacciones asociadas.	

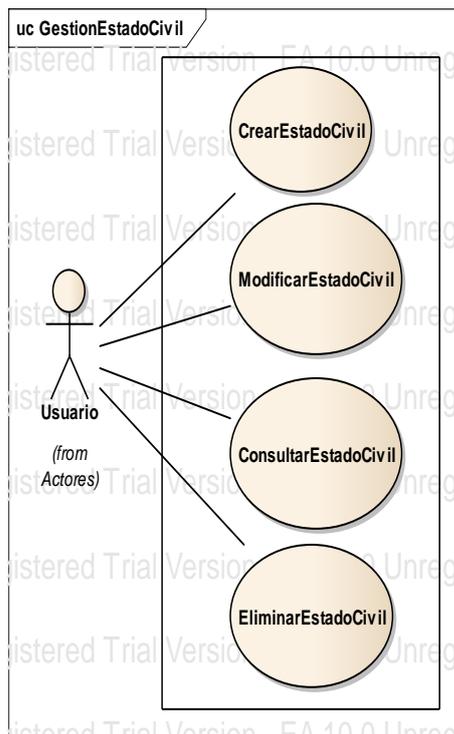


Figura B. 12 Caso de Uso Gestionar Estado Civil

B.2.13. Caso de Uso CrearRaza

Sub Modulo Secretaria	
A.2.18 Gestión Raza	
Caso de Uso	CrearRaza
Actor	Administrador
Tipo	
Descripción	Se crea el tipo de raza
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado una raza.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema ingresará el nombre de la raza que desea crear.	Si el nombre de la raza ya existe, un mensaje de error se desplegará indicando que esa raza ya se encuentra creada. Si el nombre de la raza no existe, entonces la raza se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las razas.	

B.2.14. Caso de Uso ActualizarRaza

Sub Modulo Secretaria	
A.2.18 Gestión Raza	
Caso de Uso	ActualizarRaza
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización de la raza.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado la raza.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema actualizará el nombre de la raza seleccionado.	Si el nombre de la raza ya existe, un mensaje de error se desplegará indicando que ese registro ya se encuentra creado. Si el nombre de la raza no existe, entonces el registro se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las razas.	

B.2.15. Caso de Uso ConsultarRaza

Sub Modulo Secretaria	
A.2.18 Gestión Raza	
Caso de Uso	ConsultarRaza
Actor	Administrador
Tipo	
Descripción	Se realiza consulta de raza.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado una raza.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema consultará el nombre o código de la raza.	Si el nombre o código de la raza existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código de la raza no existe, entonces la grilla se mostrará vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las razas.	

B.2.16. Caso de Uso EliminarRaza

Sub Modulo Secretaria	
A.2.18 Gestión Raza	
Caso de Uso	EliminarRaza
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de raza.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado una raza.
Acción del Actor	Respuesta del Sistema
El Administrador del sistema eliminará una raza.	Si el código de la raza existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese registro tiene transacciones. Si el código de la raza existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las razas. No se podrá eliminar una raza si tiene transacciones asociadas.	

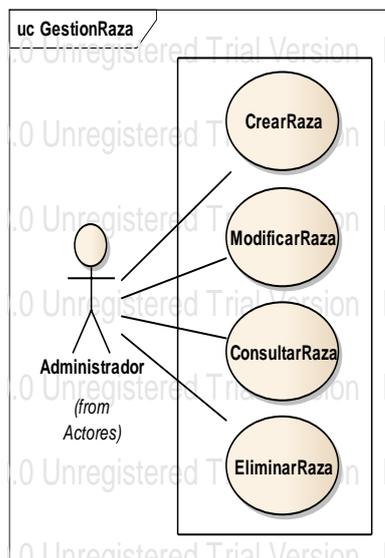


Figura B. 13 Caso de Uso Gestionar Raza

B.2.17. Caso de Uso CrearReligion

Sub Modulo Secretaria	
A.2.19 Gestión Religión	
Caso de Uso	CrearReligion
Actor	Secretaria
Tipo	
Descripción	Se realiza la creación de religión
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado una religión.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre de la religión que desea crear.	Si el nombre de la religión ya existe, un mensaje de error se desplegará indicando que esa religión ya se encuentra creada. Si el nombre de la religión no existe, entonces la religión se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las religiones.	

B.2.18. Caso de Uso ActualizarReligion

Sub Modulo Secretaria	
A.2.19 Gestión Religión	
Caso de Uso	ActualizarReligion
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización de la religión.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado una religión.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre de la religión seleccionado.	Si el nombre de la religión ya existe, un mensaje de error se desplegará indicando que esa religión ya se encuentra creada. Si el nombre de la religión no existe, entonces el registro se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las religiones.	

B.2.19. Caso de Uso ConsultarReligion

Sub Modulo Secretaria	
A.2.19 Gestión Religión	
Caso de Uso	ConsultarReligion
Actor	Administrador
Tipo	
Descripción	Se realiza consulta de religión-
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado una religión.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código de la religión.	Si el nombre o código de la religión existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código de la religión no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las religiones.	

B.2.20. Caso de Uso EliminarReligion

Sub Modulo Secretaria	
A.2.19 Gestión Religión	
Caso de Uso	EliminarReligion
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de religión.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado una religión.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará una religión.	Si el código de la religión existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que esa religión tiene transacciones. Si el código de la religión existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	Ninguna
Observaciones	
El Administrador es el responsable de la gestión de las religiones. No se podrá eliminar una religión si tiene transacciones asociadas.	

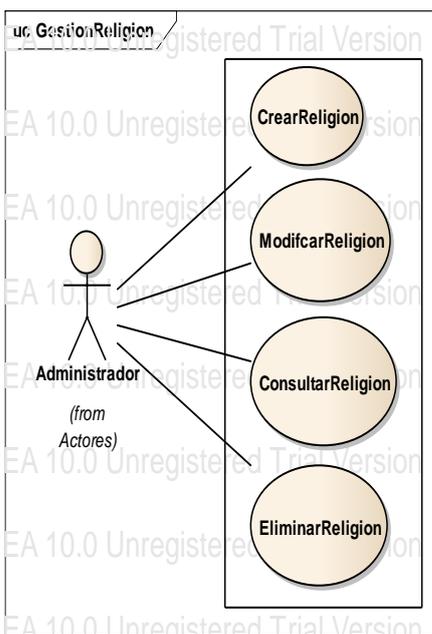


Figura B. 14 Caso de Uso Gestionar Religión

B.2.21. Caso de Uso CrearTipoSangre

Sub Modulo Secretaria	
A.2.22 Gestión Tipo Sangre	
Caso de Uso	CrearTipoSangre
Actor	Administrador
Tipo	
Descripción	Se crea el tipo de sangre.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un tipo de sangre.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del tipo de sangre que desea crear.	Si el nombre del tipo de sangre ya existe, un mensaje de error se desplegará indicando que ese tipo de sangre ya se encuentra creado. Si el nombre del tipo de sangre no existe, entonces el tipo de sangre se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los tipos de sangre.	

B.2.22. Caso de uso ActualizarTipoSangre

Sub Modulo Secretaria	
A.2.22 Gestión Tipo Sangre	
Caso de Uso	ActualizarTipoSangre
Actor	Administrador
Tipo	
Descripción	Se actualiza el tipo de sangre.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un tipo de sangre.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del tipo de sangre seleccionado.	Si el nombre del tipo de sangre ya existe, un mensaje de error se desplegará indicando que ese tipo de sangre ya se encuentra creado. Si el nombre del tipo de sangre no existe, entonces el registro se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los tipos de sangre.	

B.2.23. Caso de Uso ConsultarTipoSangre

Sub Modulo Secretaria	
A.2.22 Gestión Tipo Sangre	
Caso de Uso	ConsultarTipoSangre
Actor	Administrador
Tipo	
Descripción	Se consulta el tipo de sangre.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un tipo de sangre.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del tipo de sangre.	Si el nombre o código del tipo de sangre existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del tipo de sangre no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los tipos de sangre	

B.2.24. Caso de Uso EliminarTipoSangre

Sub Modulo Secretaria	
A.2.22 Gestión Tipo Sangre	
Caso de Uso	EliminarTipoSangre
Actor	Administrador
Tipo	
Descripción	Se elimina el tipo de sangre.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un tipo de sangre.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un tipo de sangre.	Si el código del tipo de sangre existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese tipo de sangre tiene transacciones. Si el código del tipo de sangre existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los tipos de sangre. No se podrá eliminar un tipo de sangre si tiene transacciones asociadas.	

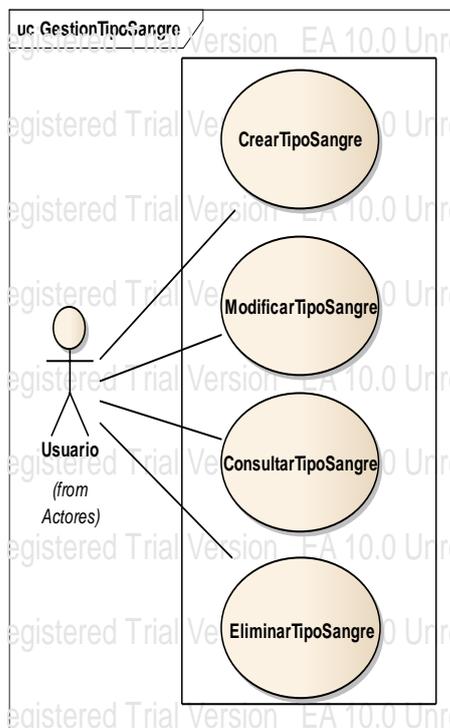


Figura B. 15 Caso de Uso Gestionar Tipo Sangre

B.2.25. Caso de Uso CrearPais

Sub Modulo Secretaria	
A.2.7 Gestión País	
Caso de Uso	CrearPais
Actor	Administrador
Tipo	
Descripción	Se crea el país
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un país.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del país que desea crear.	Si el nombre del país ya existe, un mensaje de error se desplegará indicando que ese país ya se encuentra creado. Si el nombre del país no existe, entonces el país se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los países.	

B.2.26. Caso de Uso ActualizarPais

Sub Modulo Secretaria	
A.2.7 Gestión País	
Caso de Uso	ActualizarPais
Actor	Administrador
Tipo	
Descripción	Se actualiza el país.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un país.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del país seleccionado.	Si el nombre del país ya existe, un mensaje de error se desplegará indicando que ese país ya se encuentra creado. Si el nombre del país no existe, entonces el país se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los países.	

B.2.27. Caso de uso ConsultarPais

Sub Modulo Secretaria	
A.2.7 Gestión País	
Caso de Uso	ConsultarPais
Actor	Administrador
Tipo	
Descripción	Se realiza la consulta del país
Precondición	Se ha consultado un país.
Post Condición	Se ha consultado un país.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del país.	Si el nombre o código del país existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del país no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los países.	

B.2.28. Caso de Uso EliminarPais

Sub Modulo Secretaria	
A.2.7 Gestión País	
Caso de Uso	EliminarPais
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de país.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un país.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un país.	Si el código del país existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese país tiene transacciones. Si el código del país existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	Ninguna
Observaciones	
El Administrador es el responsable de la gestión de los países.	

No se podrá eliminar un país si tiene transacciones asociadas.

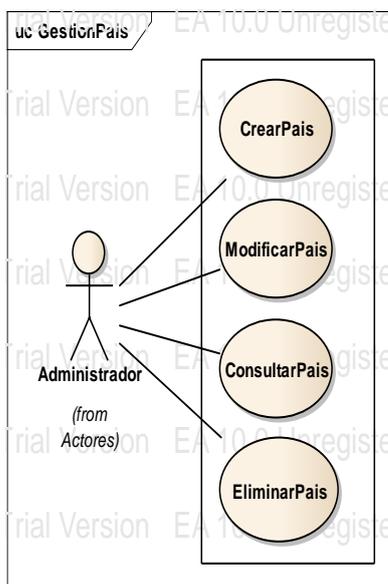


Figura B. 16 Caso de Uso Gestionar País

B.2.29. Caso de Uso CrearProvincia

Sub Modulo Secretaria	
A.2.8 Gestión Provincia	
Caso de Uso	CrearProvincia
Actor	Administrador
Tipo	
Descripción	Se realiza creación de provincia.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado una provincia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre de la provincia que desea crear.	Si el nombre de la provincia ya existe, un mensaje de error se desplegará indicando que esa provincia ya se encuentra creada. Si el nombre de la provincia no existe, entonces la provincia se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las provincias.	

B.2.30. Caso de Uso ActualizarProvincia

Sub Modulo Secretaria	
A.2.8 Gestión Provincia	
Caso de Uso	ActualizarProvincia
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización de provincia.
Precondición	El Administrador ha ingresado al sistema
Post Condición	Se ha actualizado una provincia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre de la provincia seleccionado.	Si el nombre de la provincia ya existe, un mensaje de error se desplegará indicando que esa provincia ya se encuentra creada. Si el nombre de la provincia no existe, entonces la provincia se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las provincias.	

B.2.31. Caso de Uso ConsultarProvincia

Sub Modulo Secretaria	
A.2.8 Gestión Provincia	
Caso de Uso	ConsultarProvincia
Actor	Administrador
Tipo	
Descripción	Se realiza la consulta de provincia.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado una provincia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código de la provincia.	Si el nombre o código de la provincia existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código de la provincia no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las provincias.	

B.2.32. Caso de Uso EliminarProvincia

Sub Modulo Secretaria	
A.2.8 Gestión Provincia	
Caso de Uso	EliminarProvincia
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de provincia-
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado una provincia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará una provincia.	Si el código de la provincia existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que esa provincia tiene transacciones. Si el código de la provincia existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las provincias. No se podrá eliminar una provincia si tiene transacciones asociadas.	

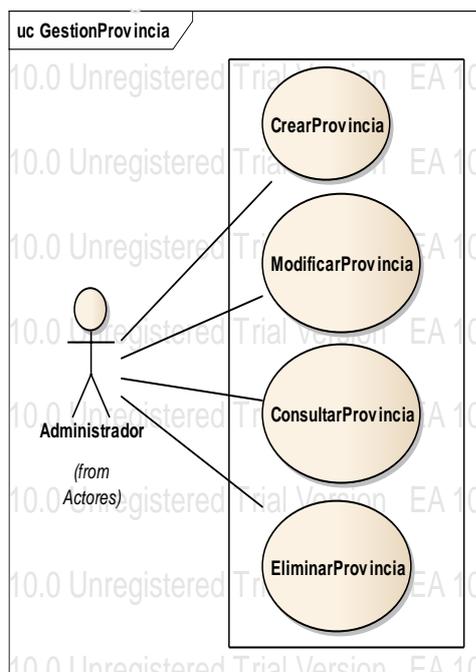


Figura B. 17 Caso de Uso Gestionar Provincia

B.2.33. Caso de Uso CrearCanton

Sub Modulo Secretaria	
A.2.9 Gestión Cantón	
Caso de Uso	CrearCanton
Actor	Administrador
Tipo	
Descripción	Se realiza la creación de cantón.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un cantón.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del cantón que desea crear.	Si el nombre del cantón ya existe, un mensaje de error se desplegará indicando que ese cantón ya se encuentra creado. Si el nombre del cantón no existe, entonces el cantón se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los cantones.	

B.2.34. Caso de Uso ActualizarCanton

Sub Modulo Secretaria	
A.2.9 Gestión Cantón	
Caso de Uso	ActualizarCanton
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización del cantón.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un cantón.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del cantón seleccionado.	Si el nombre del cantón ya existe, un mensaje de error se desplegará indicando que ese cantón ya se encuentra creado. Si el nombre del cantón no existe, entonces el cantón se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los cantones.	

B.2.35. Caso de Uso EliminarCanton

Sub Modulo Secretaria	
A.2.9 Gestión Cantón	
Caso de Uso	EliminarCanton
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de cantón.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un cantón.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un cantón.	Si el código del cantón existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese cantón tiene transacciones. Si el código del cantón existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los cantones. No se podrá eliminar un cantón si tiene transacciones asociadas.	

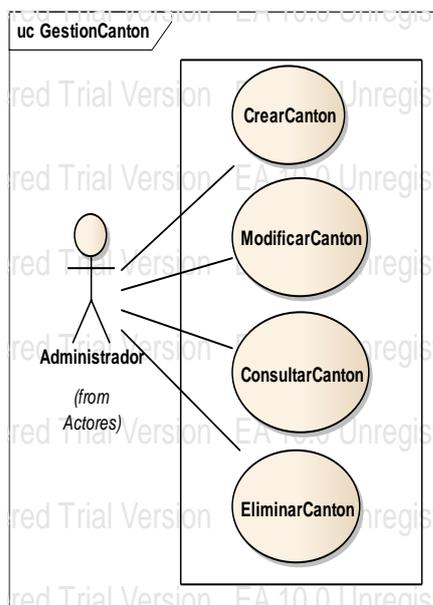


Figura B. 18 Caso de Uso Gestionar Cantón

B.2.36. Caso de Uso CrearParroquia

Sub Modulo Secretaria	
A.2.10 Gestión Parroquia	
Caso de Uso	CrearParroquia
Actor	Administrador
Tipo	
Descripción	Se realiza la creación de parroquia.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado una parroquia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre de la parroquia que desea crear.	Si el nombre de la parroquia ya existe, un mensaje de error se desplegará indicando que esa parroquia ya se encuentra creada. Si el nombre de la parroquia no existe, entonces la parroquia se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las parroquias.	

B.2.37. Caso de Uso ActualizarParroquia

Sub Modulo Secretaria	
A.2.10 Gestión Parroquia	
Caso de Uso	ActualizarParroquia
Actor	Administrador
Tipo	
Descripción	Se realiza la actualización de parroquia-
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado una parroquia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre de la parroquia seleccionada.	Si el nombre de la parroquia ya existe, un mensaje de error se desplegará indicando que esa parroquia ya se encuentra creada. Si el nombre de la parroquia no existe, entonces la parroquia se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las parroquias.	

B.2.38. Caso de Uso ConsultarParroquia

Sub Modulo Secretaria	
A.2.10 Gestión Parroquia	
Caso de Uso	ConsultarParroquia
Actor	Administrador
Tipo	
Descripción	Se realiza la consulta de parroquia.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado una parroquia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código de la parroquia.	Si el nombre o código de la parroquia existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código de la parroquia no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las parroquias.	

B.2.39. Caso de Uso EliminarParroquia

Sub Modulo Secretaria	
A.2.10 Gestión Parroquia	
Caso de Uso	EliminarParroquia
Actor	Administrador
Tipo	
Descripción	Se realiza la eliminación de la parroquia.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado una parroquia.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará una parroquia.	Si el código de la parroquia existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que esa parroquia tiene transacciones. Si el código de la parroquia existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	Ninguna
Observaciones	
El Administrador es el responsable de la gestión de las parroquias. No se podrá eliminar una parroquia si tiene transacciones asociadas.	

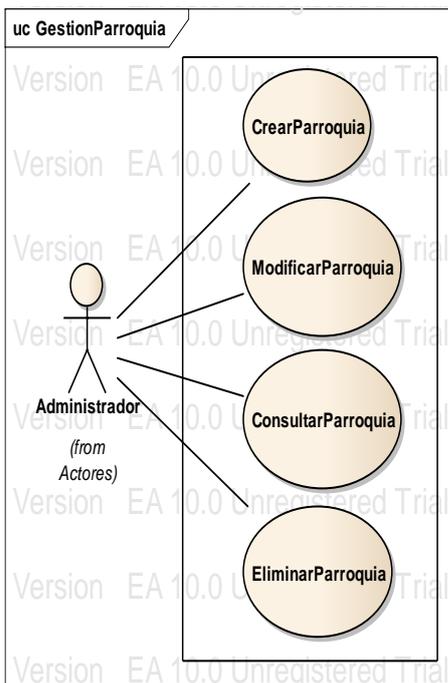


Figura B. 19 Caso de Uso Gestionar Parroquia

B.2.40. Caso de Uso CrearBarrio

Sub Modulo Secretaria	
A.2.11 Gestión Barrio	
Caso de Uso	CrearBarrio
Actor	Administrador
Tipo	
Descripción	Se realiza la creación de barrio.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un barrio.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del barrio que desea crear.	Si el nombre del barrio ya existe, un mensaje de error se desplegará indicando que ese barrio ya se encuentra creado. Si el nombre del barrio no existe, entonces el barrio se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los barrios.	

B.2.41. Caso de Uso ActualizarBarrio

Sub Modulo Secretaria	
A.2.11 Gestión Barrio	
Caso de Uso	ActualizarBarrio
Actor	Administrador
Tipo	
Descripción	Se realiza actualización de barrio.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un barrio.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del barrio seleccionado.	Si el nombre del barrio ya existe, un mensaje de error se desplegará indicando que ese barrio ya se encuentra creado. Si el nombre del barrio no existe, entonces el barrio se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los barrios.	

B.2.42. Caso de Uso ConsultarBarrio

Sub Modulo Secretaria	
A.2.11 Gestión Barrio	
Caso de Uso	ConsultarBarrio
Actor	Administrador
Tipo	
Descripción	Se realiza la consulta del barrio.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un barrio.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del barrio.	Si el nombre o código del barrio existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del barrio no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los barrios.	

B.2.43. Caso de Uso EliminarBarrio

Sub Modulo Secretaria	
A.2.11 Gestión Barrio	
Caso de Uso	EliminarBarrio
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de barrio.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un barrio.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un barrio.	Si el código del barrio existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese barrio tiene transacciones. Si el código del barrio existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el barrio.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los barrios. No se podrá eliminar un barrio si tiene transacciones asociadas.	

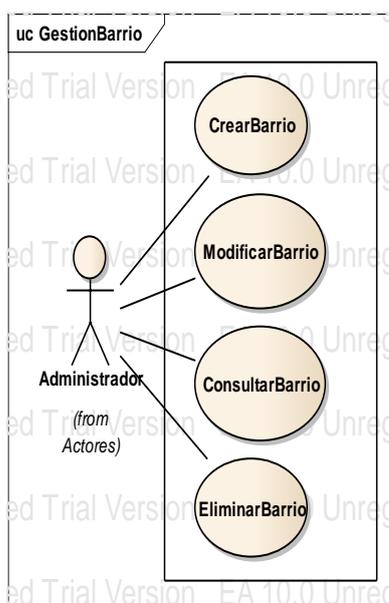


Figura B. 20 Caso de Uso Gestionar Barrio

B.2.44. Caso de Uso CrearFamiliar

Sub Modulo Secretaria	
Gestión Familiar	
Caso de Uso	CrearFamiliar
Actor	Administrador
Tipo	
Descripción	Se realiza creación de un familiar.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un familiar.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del familiar que desea crear.	Si el nombre del familiar ya existe, un mensaje de error se desplegará indicando que ese familiar ya se encuentra creado. Si el nombre del familiar no existe, entonces el familiar se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los familiares.	

B.2.45. Caso de Uso ActualizarFamiliar

Sub Modulo Secretaria	
Gestión Familiar	
Caso de Uso	ActualizarFamiliar
Actor	Administrador
Tipo	
Descripción	Se realiza actualización de un familiar.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un familiar.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del familiar seleccionado.	Si el nombre del perfil ya existe, un mensaje de error se desplegará indicando que ese familiar ya se encuentra creado. Si el nombre del familiar no existe, entonces el familiar se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los familiares.	

B.2.46. Caso de Uso ConsultaFamiliar

Sub Modulo Secretaria	
Gestión Familiar	
Caso de Uso	ConsultarFamiliar
Actor	Administrador
Tipo	
Descripción	Se realiza consulta de un familiar.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un familiar.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del familiar.	Si el nombre o código del familiar existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del familiar no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los familiares.	

B.2.47. Caso de Uso EliminarFamiliar

Sub Modulo Secretaria	
Gestión Familiar	
Caso de Uso	EliminarFamiliar
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de familiar.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un familiar.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un familiar.	Si el código del familiar existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese familiar tiene transacciones. Si el código del familiar existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los familiares. No se podrá eliminar un familiar si tiene transacciones asociadas.	

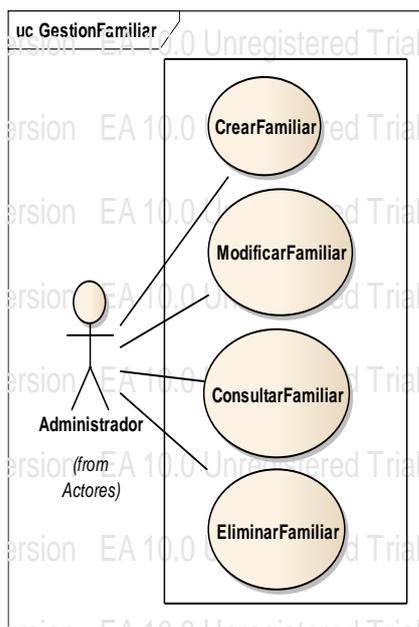


Figura B. 21 Caso de Uso Gestionar Familiar

B.2.48. Caso de Uso CrearParentesco

Sub Modulo Secretaria	
A.2.17 Gestión Parentesco	
Caso de Uso	CrearParentesco
Actor	Administrador
Tipo	
Descripción	Se realiza creación de parentesco.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un parentesco.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del parentesco que desea crear.	Si el nombre del parentesco ya existe, un mensaje de error se desplegará indicando que ese parentesco ya se encuentra creado. Si el nombre del parentesco no existe, entonces el parentesco se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los parentescos.	

B.2.49. Caso de Uso ActualizarParentesco

Sub Modulo Secretaria	
A.2.17 Gestión Parentesco	
Caso de Uso	ActualizarParentesco
Actor	Administrador
Tipo	
Descripción	Se realiza actualización de parentesco.
Precondición	El administrador ha ingresado al sistema
Post Condición	Se ha actualizado un parentesco.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizara el nombre del parentesco seleccionado.	Si el nombre del parentesco ya existe, un mensaje de error se desplegara indicando que ese parentesco ya se encuentra. Si el parentesco no existe, entonces el parentesco se actualizara en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los parentescos.	

B.2.50. Caso de Uso ConsultarParentesco

Sub Modulo Secretaria	
A.2.17 Gestión Parentesco	
Caso de Uso	ConsultarParentesco
Actor	Administrador
Tipo	
Descripción	Se realiza consulta de parentesco.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado un parentesco.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código del parentesco.	Si el nombre o código del parentesco existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del parentesco no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los parentescos.	

B.2.51. Caso de Uso EliminarParentesco

Sub Modulo Secretaria	
A.2.17 Gestión Parentesco	
Caso de Uso	EliminarParentesco
Actor	Administrador, secretaria, jefe académico
Tipo	
Descripción	Gestión Parentesco
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un parentesco.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un parentesco.	Si el código del parentesco existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese parentesco tiene transacciones. Si el código del parentesco existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los parentescos. No se podrá eliminar un parentesco si tiene transacciones asociadas.	

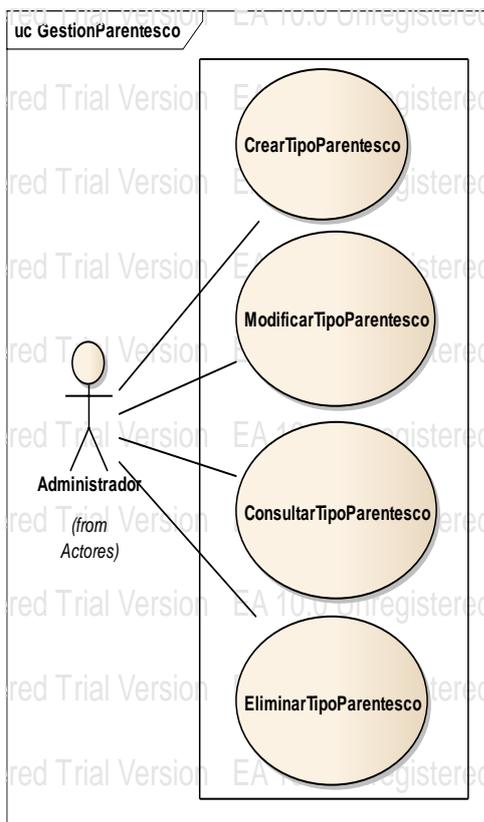


Figura B. 22 Caso de Uso Gestionar Parentesco

B.2.52. Caso de Uso CrearIdioma

Sub Modulo Secretaria	
A.2.14 Gestión Idioma	
Caso de Uso	CrearIdioma
Actor	Administrador
Tipo	
Descripción	Se realiza la creación del idioma.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un idioma.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre del idioma que desea crear.	Si el nombre del idioma ya existe, un mensaje de error se desplegará indicando que ese idioma ya se encuentra creado. Si el nombre del idioma no existe, entonces el idioma se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los idiomas.	

B.2.53. Caso de Uso ActualizarIdioma

Sub Modulo Secretaria	
A.2.14 Gestión Idioma	
Caso de Uso	ActualizarIdioma
Actor	Administrador
Tipo	
Descripción	Se realiza actualización del idioma.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado un idioma.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre del idioma seleccionado.	Si el nombre del idioma ya existe, un mensaje de error se desplegará indicando que ese idioma ya se encuentra creado. Si el nombre del idioma no existe, entonces el idioma se actualizará en el sistema.
Excepciones	Ninguna
Observaciones	
El Administrador es el responsable de la gestión de los idiomas.	

B.2.54. Caso de Uso EliminarIdioma

Sub Modulo Secretaria	
A.2.14 Gestión Idioma	
Caso de Uso	EliminarIdioma
Actor	Administrador
Tipo	
Descripción	Se realiza eliminación de idioma.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado un idioma.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará un idioma.	Si el código del idioma existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese idioma tiene transacciones. Si el código del idioma existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Observaciones	
El Administrador es el responsable de la gestión de los idiomas. No se podrá eliminar un idioma si tiene transacciones asociadas.	

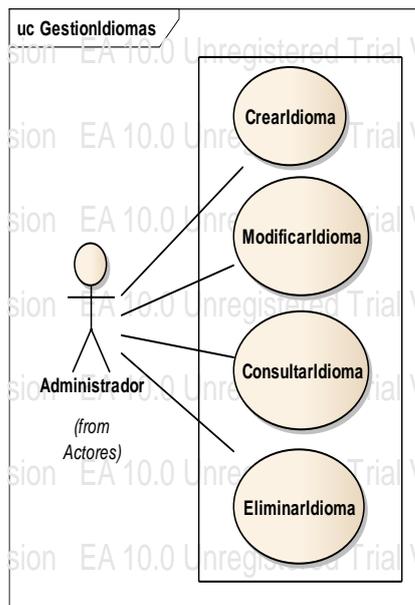


Figura B. 23 Caso de Uso Gestión Idioma

B.2.55. Caso de Uso CrearInstitucion

Sub Modulo Secretaria	
A.2.16 Gestión Instrucción	
Caso de Uso	CrearInstrucción
Actor	Administrador
Tipo	
Descripción	Se realiza creación de instrucción.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado una instrucción.
Acción del Actor	Respuesta del Sistema
El administrador del sistema ingresará el nombre de la instrucción que desea crear.	Si el nombre de la instrucción ya existe, un mensaje de error se desplegará indicando que esa instrucción ya se encuentra creada. Si el nombre de la instrucción no existe, entonces la instrucción se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las instrucciones.	

B.2.56. Caso de Uso ActualizarInstruccion

Sub Modulo Secretaria	
A.2.16 Gestión Instrucción	
Caso de Uso	ActualizarInstruccion
Actor	Administrador, secretaria, jefe académico
Tipo	
Descripción	Se realiza actualización de instrucción.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha actualizado una instrucción.
Acción del Actor	Respuesta del Sistema
El administrador del sistema actualizará el nombre de la instrucción seleccionada.	Si el nombre de la instrucción ya existe, un mensaje de error se desplegará indicando que esa instrucción ya se encuentra creada. Si el nombre de la instrucción no existe, entonces la instrucción se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las instrucciones.	

B.2.57. Caso de Uso ConsultarInstruccion

Sub Modulo Secretaria	
A.2.16 Gestión Instrucción	
Caso de Uso	ConsultarInstruccion
Actor	Administrador, secretaria, jefe académico
Tipo	
Descripción	Se realiza consulta de instrucción.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha consultado una instrucción.
Acción del Actor	Respuesta del Sistema
El administrador del sistema consultará el nombre o código de la instrucción.	Si el nombre o código de la instrucción existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código de la instrucción no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las instrucciones.	

B.2.58. Caso de Uso EliminarInstruccion

Sub Modulo Secretaria	
A.2.16 Gestión Instrucción	
Caso de Uso	EliminarInstruccion
Actor	Administrador, secretaria, jefe académico
Tipo	
Descripción	Se realiza eliminación de instrucción.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha eliminado una instrucción.
Acción del Actor	Respuesta del Sistema
El administrador del sistema eliminará una instrucción.	Si el código de la instrucción existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que esa instrucción tiene transacciones. Si el código de la instrucción existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de las instrucciones. No se podrá eliminar una instrucción si tiene transacciones asociadas.	

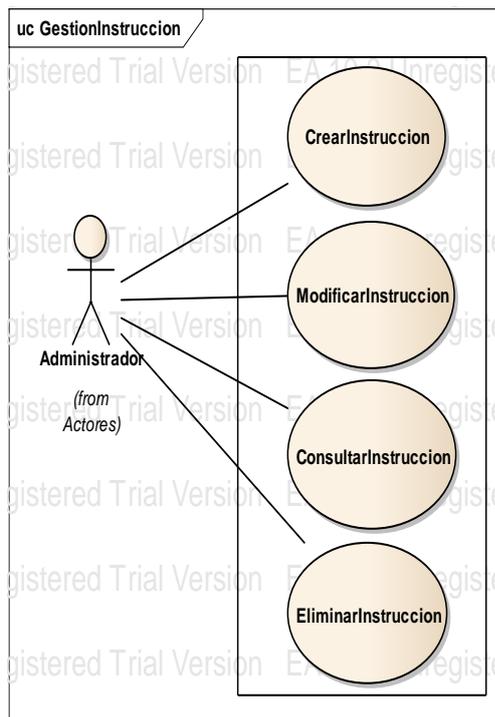


Figura B. 24 Caso de Uso Gestión Instrucción

B.2.59. Caso de Uso ConfigurarAñoLectivo

Modulo Académico	
A.2.3 Gestión Año Lectivo	
Caso de Uso	ConfigurarAñoLectivo
Actor	Administrador, Académico
Tipo	
Descripción	Configurar el año lectivo que está por iniciar
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha configurado el año lectivo.
Acción del Actor	Respuesta del Sistema
Una vez iniciado el año lectivo el usuario (jefe académico), debe crear las condiciones para que el año lectivo en vigencia sea utilizado correctamente. El usuario ingresara los datos del nuevo año lectivo, definirá las fecha de inicio y la fecha de finalización, los niveles o parciales que tendrá el nuevo año lectivo. El usuario ingresara los datos de los días de clases, de las horas de clase, los porcentajes de notas. El usuario ingresara los rangos de	El usuario guardara la información, si la información no es correcta el sistemas mostrara un mensaje caso contrario se procederá con el siguiente paso.

horas para cada hora de clase. Si los rangos no son los correctos el módulo mostrara un mensaje.	
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de configurar el año lectivo.	

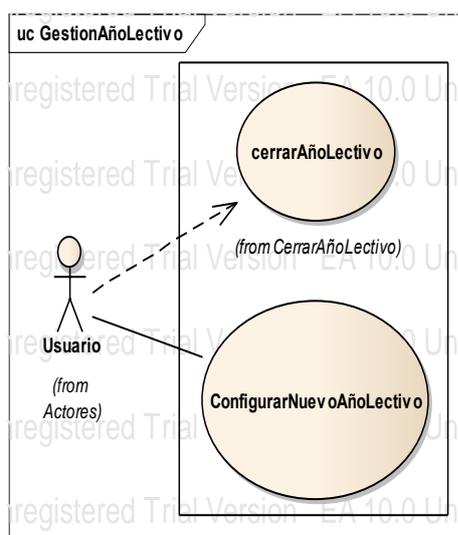


Figura B. 25 Caso de Uso Gestionar Año Lectivo

B.2.60. Caso de Uso AsignarCursoParalelo

Modulo Académico	
A.2.26 Gestión Asignar Curso Paralelo	
Caso de Uso	AsignarCursoParalelo
Actor	Administrador, jefe académico
Tipo	
Descripción	Se realiza asignación cursos paralelos.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un perfil de usuario
Acción del Actor	Respuesta del Sistema
El usuario (jefe académico), seleccionara el año lectivo creado (vigente). El usuario seleccionara el año lectivo y asociara los paralelos que ese curso podrá tener. Los nombres de los paralelos serán asociados con las letras del abecedario de una manera	El módulo desplegara los cursos creados para el año lectivo seleccionado. Si el año lectivo seleccionado no es el vigente, la información desplegada de cursos paralelos será de modo lectura.

secuencial y lógica. Los cursos – paralelos deben ser asignados para educación general y bachillerato.	
Excepciones	
Ninguna	
Observaciones	
El Administrador es el responsable de la gestión de los perfiles. No se puede crear más de un perfil con el mismo nombre.	

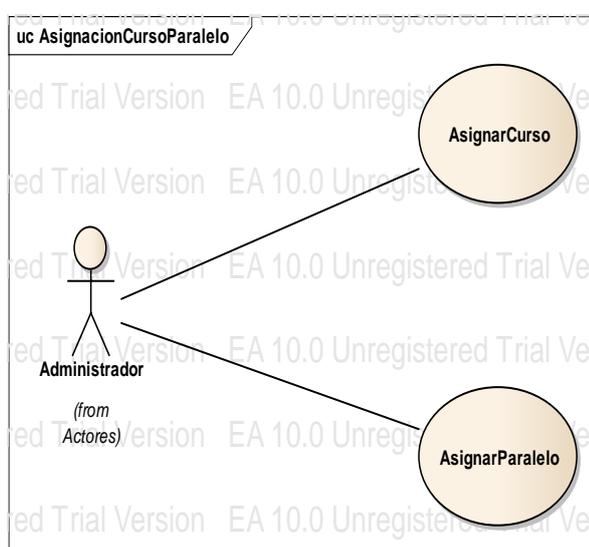


Figura B. 26 Caso de Uso Asignar Curso Paralelo

B.2.61. Caso de Uso AsignarParaleloMateria

Modulo Académico	
A.2.26 Gestión Curso Paralelo	
Caso de Uso	AsignarCursoParaleloMateria
Actor	Administrador, jefe académico
Tipo	
Descripción	Se realiza asignación de materias a los docentes por curso paralelo-
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha creado un curso-paralelo-materia.
Acción del Actor	Respuesta del Sistema
El usuario (jefe académico), seleccionara el año lectivo creado (vigente). El usuario seleccionara el curso-	El módulo desplegara los cursos - paralelos creados para el año lectivo seleccionado.

<p>paralelo y asociara las materias y docentes para ese curso-paralelo. Las materia, curso paralelo, docentes deben ser asignados para educación general y bachillerato. Si el año lectivo seleccionado no es el vigente, la información desplegada será de modo lectura. La información de otros años lectivos es histórica y es considerada solo de lectura.</p>	<p>El módulo validara y guardara la información.</p>
<p>Excepciones</p>	
<p>Ninguna</p>	
<p>Observaciones</p>	
<p>El usuario (jefe académico) es el responsable de la asignación del curso-paralelo-materia.</p>	

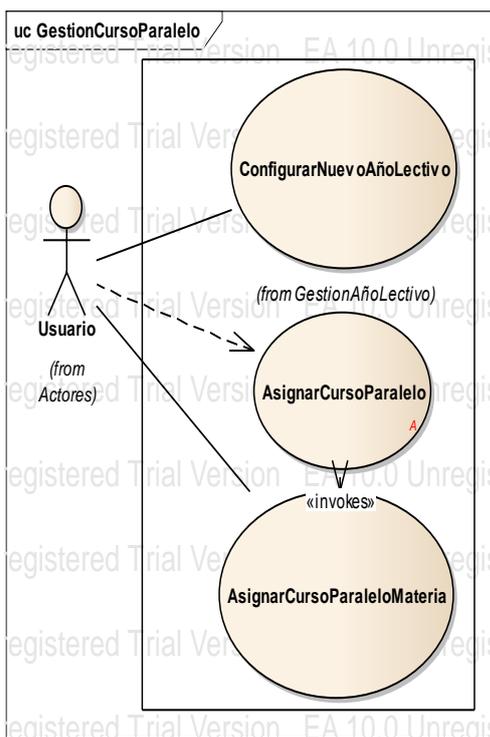


Figura B. 27 Caso de Uso Gestionar Curso Paralelo

B.2.62. Caso de Uso AsignarEstudianteAñoLectivo

Modulo Académico	
A.2.27 Gestión Estudiante Año Lectivo	
Caso de Uso	AsignarEstudianteAñoLectivo(Matricula)
Actor	Administrador, secretaria, jefe académico
Tipo	
Descripción	Realizar la matrícula de un estudiante a un curso paralelo en el año lectivo
Precondición	El Administrador ha ingresado al sistema. Una vez finalizado la asignación de materia- curso-paralelo-docente, los usuarios (jefe académico, secretaria), asigna a los estudiantes al momento de la matrícula.
Post Condición	Se ha asignado un estudiante a un año lectivo.
Acción del Actor	Respuesta del Sistema
El usuario ingresa diferentes variables para la búsqueda del estudiante. Este puede ser por cedula, apellidos, nombres o código de estudiante. El usuario selecciona al estudiante. El usuario verifica el paso de año del estudiante y asigna el curso paralelo al cual fue asignado.	El módulo devuelve los resultados al usuario. El estudiante es matriculado.
Excepciones	
Ninguna	
Observaciones	
Los usuarios (jefe académico, secretaria) son responsables de la asignación estudiante - año lectivo.	

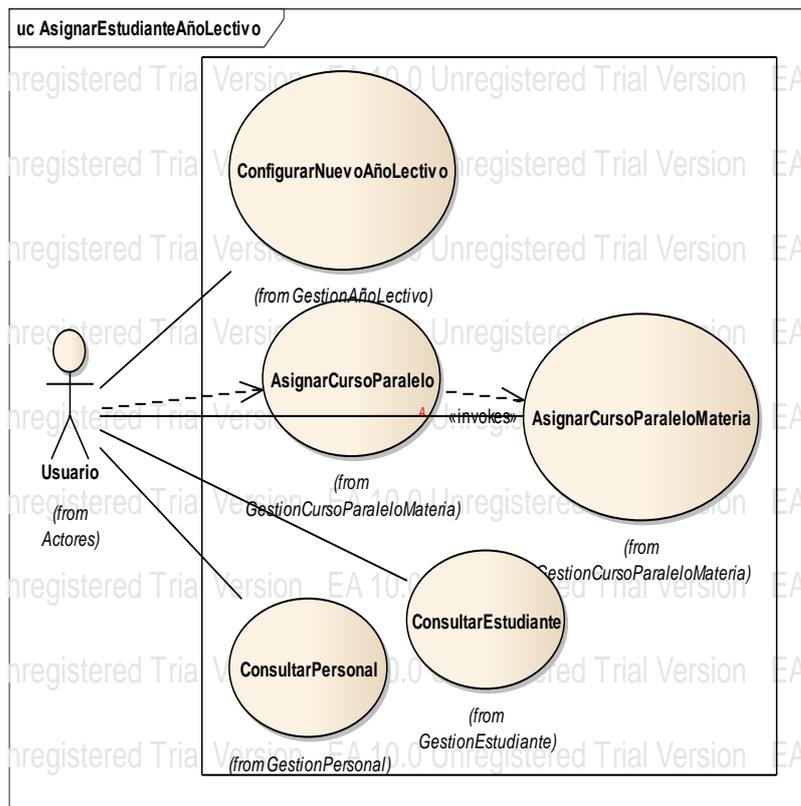


Figura B. 28 Caso de Uso Asignar Estudiante Año Lectivo

B.2.63. Caso de Uso GenerarHorarioClase

Modulo Académico	
A.2.29 Gestión Horario Clase	
Caso de Uso	GenerarHorarioClase
Actor	Administrador
Tipo	
Descripción	Generar el horario de clase para los cursos paralelos del año lectivo vigente
Precondición	El Administrador ha ingresado al sistema. Una vez iniciado el año lectivo el usuario (jefe académico), debe generar el horario de clase.
Post Condición	Se ha generado el horario de clases.
Acción del Actor	Respuesta del Sistema
El horario de clase será generado considerando el año lectivo vigente, las materias, cursos – paralelo y los docentes. Además, se debe configurar las horas y días de asistencia de clases. Si no existe un año lectivo vigente, no se podrá continuar con el proceso.	Si el horario de clase ya existe, un mensaje de error se desplegará indicando que ese horario de clase ya se encuentra creado. Si el horario de clase no existe, entonces el horario de clase se registra en el sistema.

Excepciones
Ninguna
Observaciones
El jefe académico es el responsable de la generación del horario de clase.

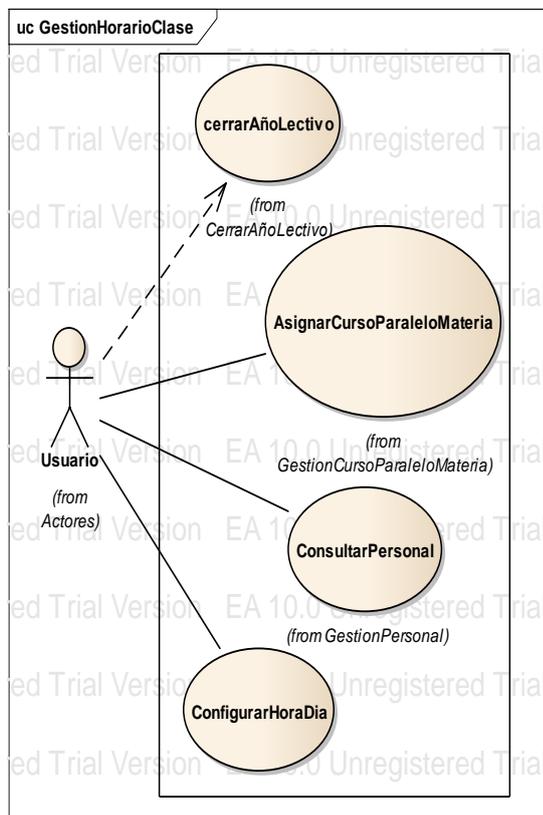


Figura B. 29 Caso de Uso Generar Horario Clase

B.2.64. Caso de Uso AsignarNotaEstudiante

Modulo Académico	
A.2.30 Gestión Asignar Nota Estudiante	
Caso de Uso	AsignarNotaEstudiante
Actor	Administrador
Tipo	
Descripción	Se realiza el paso de notas.
Precondición	El Administrador ha ingresado al sistema.
Post Condición	Se ha asignado la nota a un estudiante.
Acción del Actor	Respuesta del Sistema
El usuario (docente) asignara la nota de un estudiante, para el año lectivo vigente, a una materia determinada.	El sistema no permitirá que se ingrese valores negativos, así como valores que no se encuentren en los rangos permitidos.
El usuario deberá seleccionar el	

<p>año lectivo vigente, de ser el caso que se seleccione otro año lectivo, la información será de solo lectura.</p> <p>Una vez seleccionado el año lectivo, el usuario seleccionará el curso-paralelo, donde se visualizaran todos los estudiantes pertenecientes a ese curso-paralelo.</p> <p>El usuario ingresara la nota correspondiente al parcial. El usuario podrá ingresar la nota de uno o varios estudiantes del curso-paralelo.</p>	
Excepciones	
Ninguna	
Observaciones	
El usuario (docente) es el responsable de la asignación de notas a los estudiantes.	

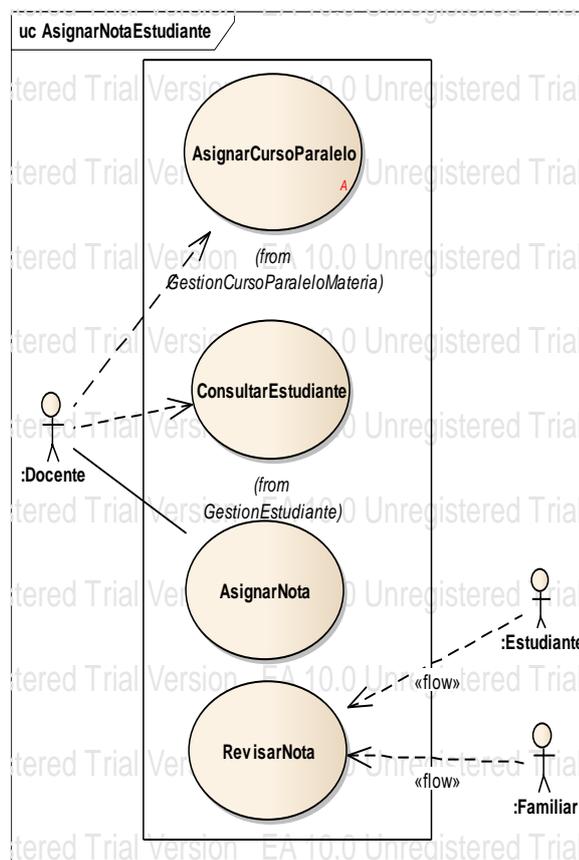


Figura B. 30 Caso de Uso Asignar Nota Estudiante

B.3. Especificación de los Casos de Uso del DOBE

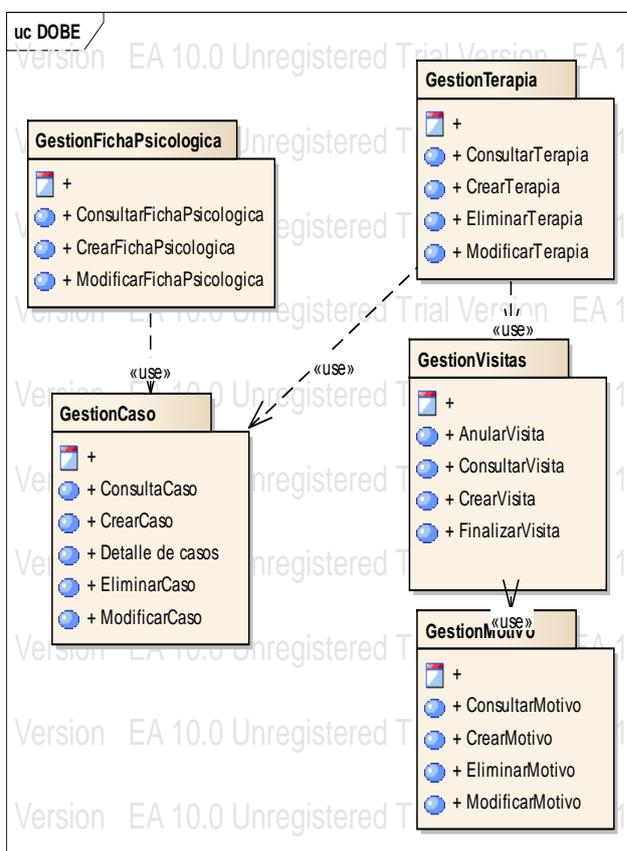


Figura B. 31 Paquete Modelo Psicología DOBE

B.3.1. Caso de Uso CrearFichaPsicologica

Módulo Dobe	
A.3.1 Gestión Ficha Psicológica	
Caso de Uso	CrearFichaPsicologica
Actor	Psicólogo
Tipo	

Descripción	Se llena la ficha psicológica del estudiante	
Precondición	El usuario ha ingresado al sistema.	
Post Condición	Se ha creado una ficha psicológica.	
Acción del Actor	Respuesta del Sistema	
El usuario (Psicólogo) ingresará la información de la ficha psicológica. El usuario selecciona el año lectivo vigente, se desplegarán los cursos paralelos, con los estudiantes pertenecientes a cada uno de ellos. El usuario seleccionará un estudiante y llenará todos los campos solicitados en la ficha.	Si todos los campos solicitados en la creación de la ficha psicológica son llenados correctamente, un mensaje se desplegará indicando que la ficha fue creada correctamente. De no ser el caso, entonces un mensaje de error será desplegado.	
Excepciones	Ninguna	
Observaciones	El usuario (Psicólogo) es el responsable de crear la ficha psicológica. La ficha psicológica no podrá ser eliminada.	

B.3.2. Caso de Uso ActualizarFichaPsicologica

Módulo Dobe		
A.3.1 Gestión Ficha Psicológica		
Caso de Uso	ActualizarFichaPsicologica	
Actor	Psicólogo	
Tipo		
Descripción	Se actualiza la ficha de los alumnos.	
Precondición	El usuario ha ingresado al sistema.	
Post Condición	Se ha actualizado una ficha psicológica.	
Acción del Actor	Respuesta del Sistema	
El usuario (Psicólogo) actualizará la ficha psicológica. El usuario puede actualizar las veces que considere necesario la ficha psicológica.	Si todos los campos solicitados en la creación de la ficha psicológica son llenados correctamente, un mensaje se desplegará indicando que la ficha fue actualizada correctamente. De no ser el caso, entonces un mensaje de error será desplegado.	
Excepciones	Ninguna	
Observaciones	El usuario (Psicólogo) es el responsable de la actualización de la ficha psicológica.	

B.3.3. Caso de Uso ConsultarFichaPsicologica

Módulo Dobe	
A.3.1 Gestión Ficha Psicológica	
Caso de Uso	ConsultarFichaPsicologica
Actor	Psicólogo
Tipo	
Descripción	Se realiza la consulta de los alumnos en la ficha.
Precondición	El usuario ha ingresado al sistema.
Post Condición	Se ha consultado una ficha psicológica.
Acción del Actor	Respuesta del Sistema
El usuario (Psicólogo) consultará ficha psicológica de un usuario.	Si el nombre o código del estudiante existe, se desplegara en una grilla los resultados de la operación. Si el nombre o código del estudiante no existe, entonces la grilla se mostrara vacía.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la consulta de la ficha psicológica	

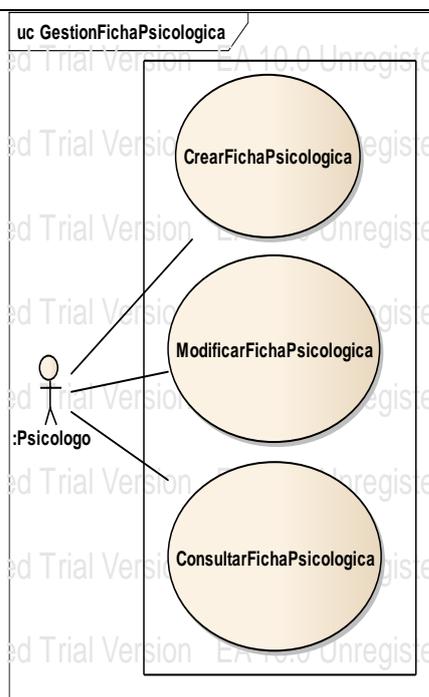


Figura B. 32 Caso de Uso Gestionar Ficha Psicológica

B.3.4. Caso de Uso CrearTerapia

Módulo Dobe	
A.3.5 Gestión Terapia	
Caso de Uso	CrearTerapia

Actor	Psicólogo	
Tipo		
Descripción	Asignar una terapia a un estudiante por medio de una caso	
Precondición	El usuario (Psicólogo) ha ingresado al sistema.	
Post Condición	Se ha creado una terapia.	
Acción del Actor	Respuesta del Sistema	
El usuario (Psicólogo) del sistema ingresará el nombre de la terapia que desea crear. El usuario (psicólogo), una vez creados los casos para los estudiantes que se encuentren en problemas deberá buscar y seleccionar el caso a tratar para asignarle una terapia adecuada y darle seguimiento, para ello podrá hacerlo de diferentes maneras: ingresando el código del estudiante, por cedula de identidad del estudiante, por nombres y apellidos o por curso- paralelo. Una vez seleccionado el estudiante se desplegara los casos asignados, seleccionara el caso y podrá asignar las terapias a cada uno de ellos.	Si el nombre de la terapia ya existe, un mensaje de error se desplegará indicando que esa terapia ya se encuentra creada. Si el nombre de la terapia no existe, entonces la terapia se registra en el sistema.	
Excepciones		
Ninguna		
Observaciones		
El usuario (Psicólogo) es el responsable de la gestión de las terapias. No se puede crear más de una terapia con el mismo nombre.		

B.3.5. Caso de Uso ActualizarTerapia

Módulo Dobe		
A.3.5 Gestión Terapia		
Caso de Uso	ActualizarTerapia	
Actor	Psicólogo	
Tipo		
Descripción	Se realiza la actualización de la terapia.	
Precondición	El usuario (Psicólogo) ha ingresado al sistema.	
Post Condición	Se ha actualizado una terapia.	
Acción del Actor	Respuesta del Sistema	
El usuario (Psicólogo) actualizará el nombre de la terapia seleccionada.	Si el nombre de la terapia ya existe, un mensaje de error se desplegará indicando que esa terapia ya se	

	encuentra creada. Si el nombre de la terapia no existe, entonces la terapia se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de las terapias. No se puede actualizar más de una terapia con el mismo nombre.	

B.3.6. Caso de Uso ConsultarTerapia

Módulo Dobe	
A.3.5 Gestión Terapia	
Caso de Uso	ConsultarTerapia
Actor	Psicólogo
Tipo	
Descripción	El usuario podrá por medio de los casos realizar la búsqueda de terapias.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha consultado una terapia.
Acción del Actor	Respuesta del Sistema
El usuario (Psicólogo) consultará el nombre o código de la terapia.	Si el nombre o código de la terapia existe, se desplegará en una grilla los resultados de la operación. Si el nombre o código de la terapia no existe, entonces la grilla se mostrara vacía.
Excepciones Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de las terapias.	

B.3.7. Caso de Uso EliminarTerapia

Módulo Dobe	
A.3.5 Gestión Terapia	
Caso de Uso	EliminarTerapia
Actor	Psicólogo
Tipo	
Descripción	Se realiza la eliminación de terapia.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha eliminado una terapia.
Acción del Actor	Respuesta del Sistema

<p>El usuario (Psicólogo) eliminará una terapia.</p>	<p>Si el código de la terapia existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que esa terapia tiene transacciones. Si el código de la terapia existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el registro.</p>
<p>Excepciones Ninguna</p>	
<p>Observaciones</p>	
<p>El usuario (Psicólogo) es el responsable de la gestión de las terapias. No se podrá eliminar una terapia si tiene transacciones asociadas.</p>	

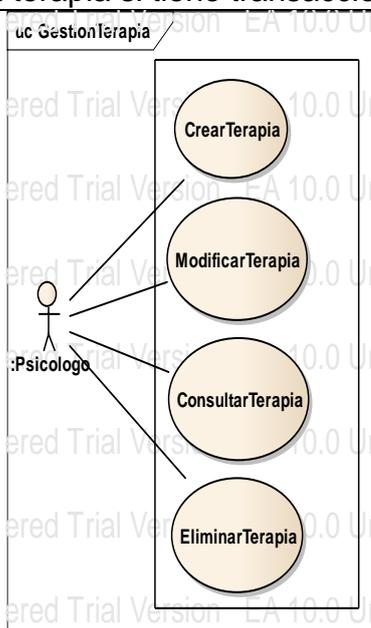


Figura B. 33 Caso de Uso Gestionar Terapia

B.3.8. Caso de Uso CrearMotivo

<p>Módulo Dobe</p>	
<p>A.3.7 Gestión Motivo</p>	
<p>Caso de Uso</p>	<p>CrearMotivo</p>
<p>Actor</p>	<p>Psicólogo</p>
<p>Tipo</p>	<p></p>
<p>Descripción</p>	<p>Se realiza la creación del motivo.</p>
<p>Precondición</p>	<p>El usuario (Psicólogo) ha ingresado al sistema.</p>
<p>Post Condición</p>	<p>Se ha creado un motivo.</p>
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>El usuario (Psicólogo) del sistema ingresará el nombre del motivo que desea crear.</p>	<p>Si el nombre del motivo ya existe, un mensaje de error se desplegará indicando que ese motivo ya se encuentra creado.</p>

	Si el nombre del motivo no existe, entonces el motivo se registra en el sistema.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de los motivos. No se puede crear más de un motivo con el mismo nombre.	

B.3.9. Caso de Uso ActualizarMotivo

Módulo Dobe	
A.3.7 Gestión Motivo	
Caso de Uso	ActualizarMotivo
Actor	Psicólogo
Tipo	
Descripción	Se realiza la actualización del motivo.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha actualizado un motivo.
Acción del Actor	Respuesta del Sistema
El usuario (Psicólogo) del sistema actualizará el nombre del motivo seleccionado.	Si el nombre del motivo ya existe, un mensaje de error se desplegará indicando que ese motivo ya se encuentra creado. Si el nombre del motivo no existe, entonces el motivo se actualizará en el sistema.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de los motivos. No se puede actualizar más de un motivo con el mismo nombre.	

B.3.10. Caso de Uso ConsultarMotivo

Módulo Dobe	
A.3.7 Gestión Motivo	
Caso de Uso	ConsultarMotivo
Actor	Psicólogo
Tipo	
Descripción	Se realiza la consultan del motivo.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha consultado un motivo.

Acción del Actor	Respuesta del Sistema
El usuario (Psicólogo) del sistema consultará el nombre o código del motivo.	Si el nombre o código del motivo existe, se desplegará en una grilla los resultados de la operación. Si el nombre o código del motivo no existe, entonces la grilla se mostrará vacía.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de los motivos.	

B.3.11. Caso de Uso EliminarMotivo

Módulo Dobe	
A.3.7 Gestión Motivo	
Caso de Uso	EliminarMotivo
Actor	Psicólogo
Tipo	
Descripción	Se realiza la eliminación del motivo.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha eliminado un motivo.
Acción del Actor	Respuesta del Sistema
El usuario (Psicólogo) del sistema eliminará un motivo.	Si el código del motivo existe y este tiene transacciones realizadas, un mensaje de error se desplegará indicando que ese motivo tiene transacciones. Si el código del motivo existe y este no tiene transacciones realizadas, un mensaje se desplegará indicando que se eliminó el motivo.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de los motivos. No se podrá eliminar un motivo si tiene transacciones asociadas.	

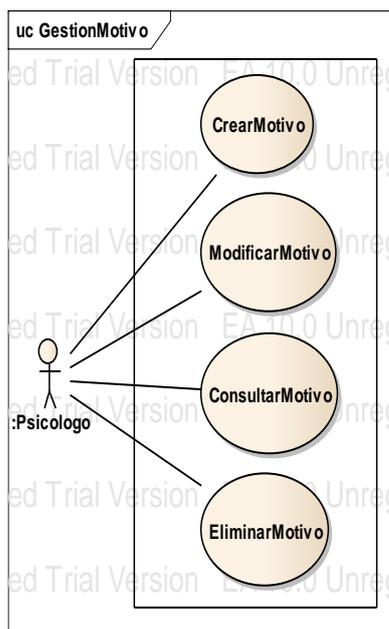


Figura B. 34 Caso de Uso Gestionar Motivo

B.3.12. Caso de Uso CrearVisita

Módulo Dobe	
A.3.6 Gestión Visita	
Caso de Uso	CrearVisita
Actor	Psicólogo
Tipo	
Descripción	Crear las visitas que los psicólogos realizan a los padres o viceversa las visitas de los padres a los estudiantes en la institución.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha creado una visita.
Acción del Actor	Respuesta del Sistema
<p>El usuario (Psicólogo) creara una visita a un estudiante o familiar. Puede seleccionar el año lectivo y visualizar los cursos- paralelos. El usuario tiene 2 opciones para asignar visitas domiciliarias a los representantes:</p> <ol style="list-style-type: none"> 1. Por medio de la búsqueda del estudiante, los criterios de búsqueda pueden ser: Apellidos, nombres, cedula, código de estudiante. 2. Seleccionara un curso-paralelo, y en el detalle seleccionara un estudiante. 	<p>El estado de la visita domiciliaria quedara en "Proceso", hasta que se realice todo el proceso y se entreviste el representante con el psicólogo.</p>

<p>Cuando seleccione el estudiante será enviado al formulario de visitas domiciliarias.</p> <p>El usuario podrá realizar varias visitas al estudiante o familiar.</p> <p>Una vez allí, el psicólogo deberá ingresar la fecha en la que se realiza la visita al domicilio, buscara al alumno y lo seleccionara.</p> <p>El psicólogo colocara la fecha de asistencia al domicilio del estudiante.</p> <p>El psicólogo debe colocar un motivo por el cual se realiza la visita domiciliaria.</p>	
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de las visitas.	

B.3.13. Caso de Uso FinalizaVisita

Módulo Dobe	
A.3.6 Gestión Visita	
Caso de Uso	FinalizarVisita
Actor	Psicólogo
Tipo	
Descripción	Finalizar la visita, El usuario (Psicólogo) finalizara una visita a un estudiante o familiar.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha finalizado una visita.
Acción del Actor	Respuesta del Sistema
<p>El usuario deberá seleccionar las visitas que se encuentran en estado en "proceso", para el estudiante deseado, una vez digitados los comentarios respectivos.</p> <p>Una vez realiza la visita domiciliaria con el representante del alumno, el psicólogo debe ingresar nuevamente a la visita domiciliaria y colocarla en estado "Finalizado", de ser el caso ingresara alguna observación.</p> <p>Una vez finalizada la visita, no se podrán realizar cambios.</p>	<p>El sistema mostrara un mensaje de visita guardada y finalizada.</p> <p>El estado de la visita será cambiado a Finalizado.</p>

Excepciones
Ninguna
Observaciones
El usuario (Psicólogo) es el responsable de la gestión de las visitas.

B.3.14. Caso de Uso AnularVisita

Módulo Dobe	
A.3.6 Gestión Visita	
Caso de Uso	AnularVisita
Actor	Psicólogo
Tipo	
Descripción	El usuario podrá anular las visitas
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha anulado una visita.
Acción del Actor	Respuesta del Sistema
El usuario (Psicólogo) anulara una visita a un estudiante o familiar.	Si la visita no se realizó el sistema cambiara el estado y colocarlo como anulada. Una vez anulada la visita, no se podrán realizar cambios.
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de las visitas.	

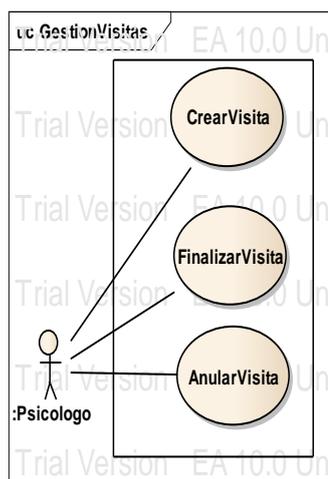


Figura B. 35 Caso de Uso Gestionar Visita

B.3.15. Caso de Uso CrearCaso

Módulo Dobe
A.3.3 Gestionar Caso

Caso de Uso	CrearCaso
Actor	Psicólogo
Tipo	
Descripción	Los usuarios crearan casos de seguimiento a los estudiantes que presenten algún tipo de problemas.
Precondición	El usuario (Psicólogo) ha ingresado al sistema.
Post Condición	Se ha creado un caso.
Acción del Actor	Respuesta del Sistema
<p>El usuario (Psicólogo), seleccionara el año lectivo y visualizar los cursos- paralelos.</p> <p>El usuario tiene 2 opciones para asignar casos:</p> <ol style="list-style-type: none"> 1. Por medio de la búsqueda del estudiante, los criterios de búsqueda pueden ser: Apellidos, nombres, cedula, código de estudiante. 2. Seleccionara un curso-paralelo, y en el detalle seleccionara un estudiante. <p>Los casos sirven para llevar un control de visitas, entrevistas, en una fecha determinada. El usuario deberá llenar el motivo por el cual se ha creado un caso al estudiante.</p>	<p>Al momento de la creación del caso el estado será en “proceso”.</p> <p>El sistemas mostrara el mensaje de guardado.</p>
Excepciones	
Ninguna	
Observaciones	
El usuario (Psicólogo) es el responsable de la gestión de los casos.	

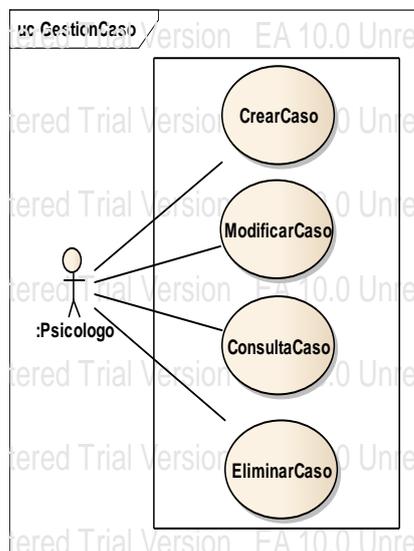


Figura B. 36 Caso de Uso Gestionar Caso

Anexo C

CD Diagrama Conceptual de la Base de Datos

Anexo D

CD Diagrama Físico de la Base de Datos

E.1.1.1. Anexo E

Pruebas Unitarias

E. PRUEBAS

E.1.Pruebas de Unidad Módulo de Administración

Las siguientes clases pertenecen al módulo de Administración

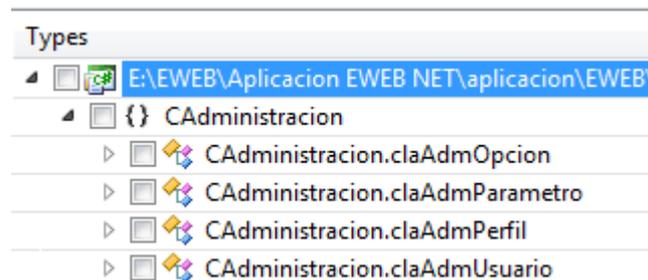


Figura E. 1 Clases de prueba Modulo Administración

Prueba Clase AdmOpcion

E.1.1.2. Método Asignar Perfil Módulo Adicional

```
[TestMethod()]
[HostType("ASP.NET")]
[AspNetDevelopmentServerHost("E:\EWEB\Aplicacion EWEB NET\aplicacion\EWEB", "/EWEB")]
[UrlToTest("http://localhost/EWEB")]
public void funAdmAsignarPerfilModuloAdicionalTest()
{
    claAdmOpcion_Accessor target = new claAdmOpcion_Accessor();
    short usuCodigo = 8991;
    short opcCodigo = 21;
    target.funAdmAsignarPerfilModuloAdicional(usuCodigo, opcCodigo);
}
```

Figura E. 2 Prueba Método Asignar Perfil Módulo Adicional

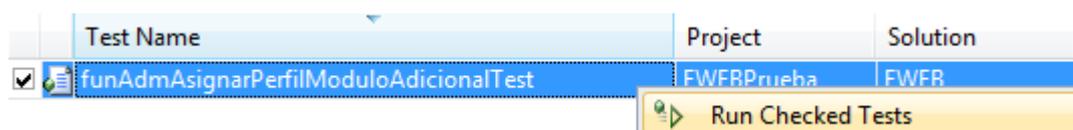


Figura E. 3 Ejecución prueba Asignar Perfil Módulo Adicional

Result	Test Name	Project	Error Message
Passed	funAdmAsignarPerfilModuloAdicionalTest	EWEBPrueba	

Figura E. 4 Resultado prueba Asignar Perfil Módulo Adicional

E.1.1.3. Método Módulo Opción Usuario

```
[TestMethod()]
[HostType("ASP.NET")]
[AspNetDevelopmentServerHost("E:\\EWEB\\Aplicacion EWEB NET\\aplicacion\\EWEB", "/EWEB")]
[UrlToTest("http://localhost/EWEB")]
public void funAdmModuloOpcionUsuarioTest()
{
    claAdmOpcion_Accessor target = new claAdmOpcion_Accessor();
    short perCodigo = 0;
    short usuCodigo = 0;
    DataSet expected = null;
    DataSet actual;
    actual = target.funAdmModuloOpcionUsuario(perCodigo, usuCodigo);
    Assert.AreEqual(expected, actual);
}
}
```

Figura E. 5 Prueba Método Opción Usuario

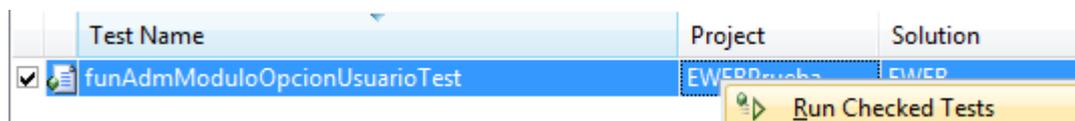


Figura E. 6 Ejecución prueba Opción Usuario

Result	Test Name	Project	Error Message
Passed	funAdmModuloOpcionUsuarioTest	EWEBPrueba	

Figura E. 7 Resultado prueba Opción Usuario

E.1.1.4. Método Obtener Perfil Opción

Result	Test Name	Project	Error Message
Passed	funAdmObtenerPerfilOpcionTest	EWEBPrueba	

Figura E. 8 Resultado prueba Obtener Perfil Opción

E.1.1.5. Método Gestión Opción Perfil

Result	Test Name	Project	Error Message
Passed	spAdmGestionOpcionPerfilTest	EWEBPrueba	

Figura E. 9 Resultado prueba Gestión Opción Perfil

Pruebas Clase Perfil

E.1.1.6. Método Gestión Perfil

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAdmGestionPerfilTest	EWEBPrueba	

Figura E. 10 Resultado prueba Método Gestión Perfil

E.1.1.7. Método Obtener Detalle Perfil

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAdmObtenerDetallePerfilTest	EWEBPrueba	

Figura E. 11 Resultado prueba Obtener Detalle Perfil

Prueba Clase Usuario

E.1.1.8. Método Buscar Usuario

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAdmBuscarUsuarioTest	EWEBPrueba	

Figura E. 12. Resultado prueba Buscar Usuario

E.1.1.9. Método Buscar Usuario por Perfil

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAdmBuscarUsuarioPorPerfilTest	EWEBPrueba	

Figura E. 13 Resultado prueba Buscar Usuario por Perfil

E.1.1.10. Método Cambiar Password

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAdmCambiarPasswordTest	EWEBPrueba	

Figura E. 14 Resultado prueba Cambiar Password

E.1.1.11. Método Cifrar Clave

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAdmCifrarClaveTest	EWEBPrueba	

Figura E. 15 Resultado prueba Cifrar Clave

E.1.1.12. Método Quitar Acceso

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
 Passed	funAdmDarQuitarAccesoTest	EWEBPrueba	

Figura E. 16 Resultado prueba Quitar Acceso

E.1.1.13. Método Gestión Usuario

```
[TestMethod()]
[HostType("ASP.NET")]
[AspNetDevelopmentServerHost("E:\\EWEB\\Aplicacion EWEB NET\\aplicacion\\EWEB", "/EWEB")]
[UrlToTest("http://localhost/EWEB")]
public void funAdmGestionUsuarioTest()
{
    claAdmUsuario_Accessor target = new claAdmUsuario_Accessor();
    char tipoOperacion = 'I';
    short usuCodigo = 1252;
    short peCodigo = 1;
    int esCodigo = null;
    int famCodigo = null;
    string usuUsername = "ocarate";
    string usuPassword = "1109610113094187236222011691145316696189251724218499123726154157176101068642552";
    short perCodigo = 1;
    char usuEstado = 'A';
    short expected = 0;
    short actual;
    actual = target.funAdmGestionUsuario(tipoOperacion, usuCodigo, peCodigo, esCodigo, famCodigo, usuUsername,
        usuPassword, perCodigo, usuEstado);
    Assert.AreEqual(expected, actual);
}
```

Figura E. 17 Prueba Método Gestión Usuario

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
 Passed	funAdmGestionUsuarioTest	EWEBPrueba	

Figura E. 18 Resultado prueba Gestión Usuario

E.1.1.14. Método Grabar Log Usuario

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
 Passed	funAdmGrabarLogUsuarioTest	EWEBPrueba	

Figura E. 19 Resultado prueba Grabar Log Usuario

E.1.1.15. Método Obtener Parámetros Seguridad

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
 Passed	funAdmObtenerParametrosSeguridadTest	EWEBPrueba	

Figura E. 20 Resultado prueba Obtener Parámetros Seguridad

E.1.1.16. Método Obtener Perfil

	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funAdmObtenerPerfilTest	EWEBPrueba	

Figura E. 21 Resultado prueba Obtener Perfil

E.1.1.17. Método Resetear Password

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funAdmResetearPasswordTest	EWEBPrueba	

Figura E. 22 Resultado prueba Resetear Password

Pruebas Unidad Módulo Académico

Prueba Clase Barrio

E.1.1.18. Método Gestión Barrio

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecGestionBarrioTest	EWEBPrueba	

Figura E. 23 Resultado prueba Gestión Barrio

Prueba Clase Cantón

E.1.1.19. Método Gestión Cantón

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecGestionCantonTest	EWEBPrueba	

Figura E. 24 Resultado prueba Gestión Cantón

E.1.1.20. Método Buscar Estudiante

```
[TestMethod()]
[HostType("ASP.NET")]
[AspNetDevelopmentServerHost("E:\\EWEB\\Aplicacion EWEB NET\\aplicacion\\EWEB", "/EWEB")]
[UrlToTest("http://localhost/EWEB")]
public void funSecBuscarEstudianteTest()
{
    claSecEstudiante_Accessor target = new claSecEstudiante_Accessor();
    string esCodigo = 1240;
    string esApellidoPaterno = "Apellidos1 Prueba";
    string esApellidoMaterno = "Apellidos2 Prueba";
    string esNombres = "Nombre1 Prueba";
    string esNombrePila = "Prueba";
    string esIdentificacion = "1715336689";
    string esSexo = "M";
    string tipsCodigo = 1;
    string razCodigo = 2;
    string relCodigo = 2;
    string estCodigo = 1;
    SqlDataReader expected = null;
    SqlDataReader actual;
    actual = target.funSecBuscarEstudiante(esCodigo, esApellidoPaterno, esApellidoMaterno,
        esNombres, esNombrePila,
        esIdentificacion, esSexo, tipsCodigo, razCodigo, relCodigo, estCodigo);
    Assert.AreEqual(expected, actual);
}
}
```

Figura E. 25 Prueba Método Buscar Estudiante

	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecBuscarEstudianteTest	EWEBPrueba	

Figura E. 26 Resultado prueba Buscar Estudiante

Prueba Clase Familia

E.1.1.21. Método Familia por estudiante

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecFamiliaPorEstudianteTest	EWEBPrueba	

Figura E. 27 Resultado prueba Familia por estudiante

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecGestionEstudianteFamiliaTest	EWEBPrueba	

Figura E. 28 Resultado prueba Gestión Estudiante por Familia

Prueba Clase SecPais

E.1.1.22. Método Gestión País

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecGestionPaisTest	EWEBPrueba	

Figura E. 29 Resultado prueba Gestión País

Prueba Clase SecParentesco

E.1.1.23. Método Gestión Parentesco

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecGestionParentescoTest	EWEBPrueba	

Figura E. 30 Resultado prueba Gestión Parentesco

Prueba Clase Parroquia

E.1.1.24. Método Gestión Parroquia

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecGestionParroquiaTest	EWEBPrueba	

Figura E. 31 Resultado prueba Gestión Parroquia

E.1.1.25. Método Buscar Personal

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funSecBuscarPersonalTest	EWEBPrueba	

Figura E. 32 Resultado prueba Buscar Personal

E.1.1.26. Método Gestión Personal

```
[TestMethod()]
[HostType("ASP.NET")]
[AspNetDevelopmentServerHost("E:\\EWEB\\Aplicacion EWEB NET\\aplicacion\\EWEB", "/EWEB")]
[UrlToTest("http://localhost/EWEB")]
public void funSecGestionPersonalTest()
{
    claSecPersonal_Accessor target = new claSecPersonal_Accessor();
    char tipoOperacion = 'I'; int peCodigo = 521; short tipsCodigo = 1;
    short razCodigo = 1; short tippCodigo = 1;
    short relCodigo = 1; short estcCodigo = 1;
    string peApellidoPaterno = "Prueba Apellido";
    string peApellidoMaterno = "Prueba Apellido";
    string peNombres = "Prueba Nombre";
    char peTipoIdentificacion = 'C'; string peIdentificacion = "1715336689";
    char peValidadaIdentificacion = 'S'; string peIess = string.Empty; string peLibretaMilitar = string.Empty;
    char peSexo = 'M'; string peColorPiel = string.Empty;
    string peColorOjos = string.Empty;
    string peColorCabello = string.Empty;
    short peEstatura = 0; short peTallaViste = 0;
    short peCalzado = 0; short peGrado = 0;
    short peArma = 0; DateTime peFechaNacimiento = "19780525";
    DateTime peFechaIngreso = new DateTime();
    short estCodigo = 0; char pePerteneceConsejo = 'N';
    short expected = 0;
    short actual;
    actual = target.funSecGestionPersonal(tipoOperacion, peCodigo, tipsCodigo, razCodigo, tippCodigo,
    relCodigo, estcCodigo, peApellidoPaterno, peApellidoMaterno, peNombres, peTipoIdentificacion,
    peIdentificacion, peValidadaIdentificacion, peIess, peLibretaMilitar, peSexo, peColorPiel,
    peColorOjos, peColorCabello, peEstatura, peTallaViste, peCalzado, peGrado, peArma,
    peFechaNacimiento, peFechaIngreso, estCodigo, pePerteneceConsejo);
    Assert.AreEqual(expected, actual);
}
}
```

Figura E. 33 Prueba Método Gestión Personal

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
Passed	funSecGestionPersonalTest	EWEBPrueba	

Figura E. 34 Resultado prueba Gestión Personal

E.1.1.27. Método Gestión Personal Capacitación

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
Passed	funSecGestionPersonalCapacitacionTest	EWEBPrueba	

Figura E. 35 Resultado prueba Gestión Personal Capacitación

Prueba Clase SecProvincia

E.1.1.28. Método Gestión Provincia

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
Passed	funSecGestionProvinciaTest	EWEBPrueba	

Figura E. 36 Resultado prueba Gestión Provincia

Prueba Clase Raza

E.1.1.29. Método Gestión Raza

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funSecGestionRazaTest	EWEBPrueba	

Figura E. 37 Resultado prueba Gestión Raza

Prueba Clase SecTipoDomicilio

E.1.1.30. Método Gestión Tipo Domicilio

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funSecGestionTipoDomicilioTest	EWEBPrueba	

Figura E. 38 Resultado prueba Gestión Tipo Domicilio

Prueba Clase Tipo Personal

E.1.1.31. Método Gestión Tipo Personal

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funSecGestionTipoPersonalTest	EWEBPrueba	

Figura E. 39 Resultado prueba Gestión Tipo Personal

Prueba Clase SecUbicacion

E.1.1.32. Método Gestión Teléfono

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funSecGestionTelefonosTest	EWEBPrueba	

Figura E. 40 Resultado prueba Gestión Teléfono

E.1.1.33. Método Gestión Ubicación

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funSecGestionUbicacionTest	EWEBPrueba	

Figura E. 41 Resultado prueba Gestión Ubicación

Prueba Clase AcaAreaAcademica

E.1.1.34. Método Gestión Área Académica

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaGestionAreaAcademicaTest	EWEBPrueba	

Figura E. 42 Resultado prueba Gestión Área Académica

Prueba Clase AcaAreaParalelo

E.1.1.35. Método Año Lectivo Curso

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaAnioLectivoCursoTest	EWEBPrueba	

Figura E. 43 Resultado prueba Año Lectivo Curso

E.1.1.36. Método Año Lectivo Paralelo Curso

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaAnioLectivoParalelosPorCursoTest	EWEBPrueba	

Figura E. 44 Resultado prueba Año Lectivo Paralelo Curso

E.1.1.37. Método Gestión Curso Paralelo

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaGestionCursoParaleloTest	EWEBPrueba	

Figura E. 45 Resultado prueba Gestión Curso Paralelo

E.1.1.38. Método Gestión Curso

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaGestionCursoTest	EWEBPrueba	

Figura E. 46 Resultado prueba Gestión Curso

E.1.1.39. Método Obtener Curso

Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	Passed	funAcaObtenerCursoTest	EWEBPrueba	

Figura E. 47 Resultado prueba Obtener Curso

E.1.1.40. Método Obtener Curso Materia

Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	Passed	funAcaObtenerCursoMateriaTest	EWEBPrueba	

Figura E. 48 Resultado prueba Obtener Curso Materia

Prueba Clase Especialidad

E.1.1.41. Método Gestión Especialidad

Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	Passed	funAcaGestionEspecialidadTest	EWEBPrueba	

Figura E. 49 Resultado prueba Gestión Especialidad

	Test Name	Project	Solution
<input checked="" type="checkbox"/>	funAcaListadoEspecialidadTest	EWEBPrueba	EWEB

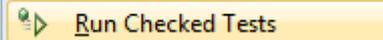


Figura E. 50 Ejecución prueba Listado Especialidad

Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	Passed	funAcaListadoEspecialidadTest	EWEBPrueba	

Figura E. 51 Resultado prueba Listado Especialidad

Prueba Clase AcaEstructuraAnio

E.1.1.42. Método Estructura Año Actual

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaEstructuraAnioActualTest	EWEBPrueba	

Figura E. 52 Resultado prueba Estructura Año Actual

E.1.1.43. Método Estructura Año Lectivo

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaEstructuraAnioLectivoTest	EWEBPrueba	

Figura E. 53 Resultado prueba Estructura Año Lectivo

E.1.1.44. Método Parcial Año Lectivo

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaParcialPorAnioLectivoTest	EWEBPrueba	

Figura E. 54 Resultado prueba Parcial Año Lectivo

E.1.1.45. Método Valor Académico Año Lectivo

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaValorAcademicaAnioLectivoTest	EWEBPrueba	

Figura E. 55 Resultado prueba Valor Académico Año Lectivo

E.1.1.46. Método Valor Conducta Año Lectivo

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaValorConductaAnioLectivoTest	EWEBPrueba	

Figura E. 56 Valor Conducta Año Lectivo

E.1.1.47. Método Fecha Año Lectivo Vigencia

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaFechaAnioLectivoVigenciaTest	EWEBPrueba	

Figura E. 57 Resultado prueba Fecha Año Lectivo Vigencia

E.1.1.48. Método Estructura Año Unifica

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaEstructuraAnioUnificaTest	EWEBPrueba	

Figura E. 58 Resultado prueba Estructura Año Unifica

E.1.1.49. Método Estructura Curso Paralelo

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaEstructuraCursoParaleloTest	EWEBPrueba	

Figura E. 59 Resultado prueba Estructura Curso Paralelo

E.1.1.50. Método Estructura Nivel

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaEstructuraNivelTest	EWEBPrueba	

Figura E. 60 Resultado prueba Estructura Nivel

E.1.1.51. Método Fuera Horario Clase

 Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaFueraHorarioClaseTest	EWEBPrueba	

Figura E. 61 Resultado prueba Fuera Horario Clase

E.1.1.52. Método Gestión Horario Clase

```
[TestMethod()]
[HostType("ASP.NET")]
[AspNetDevelopmentServerHost("E:\\EWEB\\Aplicacion EWEB NET\\aplicacion\\EWEB", "/EWEB")]
[UrlToTest("http://localhost/EWEB")]
public void funAcaGestionHorarioClaseTest()
{
    claAcaHorarioClase_Accessor target = new claAcaHorarioClase_Accessor();
    short estaCodigo = 145;
    short horCodigo = 1;
    short curpCodigo = 125;
    string materiaLunes = "41";
    string materiaMartes = string.Empty;
    string materiaMiercoles = string.Empty;
    string materiaJueves = string.Empty;
    string materiaViernes = string.Empty;
    string materiaSabado = string.Empty;
    string materiaDomingo = string.Empty;
    short expected = 0;
    short actual;
    actual = target.funAcaGestionHorarioClase(estaCodigo, horCodigo,
    curpCodigo, materiaLunes, materiaMartes, materiaMiercoles,
    materiaJueves, materiaViernes, materiaSabado, materiaDomingo);
    Assert.AreEqual(expected, actual);
}
}
```

Figura E. 62 Prueba Método Gestión Horario Clase

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
Passed	funAcaGestionHorarioClaseTest	EWEBPrueba	

Figura E. 63 Resultado prueba Gestión Horario Clase

E.1.1.53. Método Horario Clase Docente Materia

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
Passed	funAcaHorarioClaseDocenteMateriaTest	EWEBPrueba	

Figura E. 64 Resultado prueba Horario Clase Docente Materia

E.1.1.54. Método Obtener Horario Clase Curso Paralelo

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
Passed	funAcaObtenerHorarioClaseCursoParaleloTest	EWEBPrueba	

Figura E. 65 Resultado prueba Obtener Horario Clase Curso Paralelo

E.1.1.55. Método Obtener Materia Docente Curso

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funAcaObtenerMateriaDocenteCursoTest	EWEBPrueba	

Figura E. 66 Resultado prueba Obtener Materia Docente Curso

E.1.1.56. Método Verifica Configuración Hora Día

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funAcaVerificaConfiguracionHoraDiaTest	EWEBPrueba	

Figura E. 67 Resultado prueba Verifica Configuración Hora Día

Prueba Clase AcaMateriaCurso

E.1.1.57. Método Gestionar Materia Curso

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funAcaGestionMateriaCursoTest	EWEBPrueba	

Figura E. 68 Ejecución prueba Gestionar Materia Curso

Prueba Clase AcaMateria

E.1.1.58. Método Gestión Materia

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funAcaGestionMateriaTest	EWEBPrueba	

Figura E. 69 Resultado prueba Gestión Materia

Prueba Clase AcaNivel

E.1.1.59. Método Gestión Nivel

 Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funInsGestionNivelTest	EWEBPrueba	

Figura E. 70 Resultado prueba Gestión Nivel

Prueba Clase AcaNotaMateriaEstudiante

E.1.1.60. Método Estudiante Nota

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaEstudianteNotaTest	EWEBPrueba	

Figura E. 71 Ejecución prueba Estudiante Nota

E.1.1.61. Método Gestión Nota Estudiante Materia

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funAcaGestionNotaEstudianteMateriaTest	EWEBPrueba	

Figura E. 72 Resultado prueba Gestión Nota Estudiante Materia

Pruebas Unidad Módulo DOBE

Prueba Clase PsiFichaEstudiante

E.1.1.62. Método Elimina Estudiante Orienta Tarea

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funPsiEliminaEstudianteOrientaTareaTest	EWEBPrueba	

Figura E. 73 Resultado prueba Elimina Estudiante Orienta Tarea

E.1.1.63. Método Elimina Estudiante Vive con

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funPsiEliminaEstudianteViveConTest	EWEBPrueba	

Figura E. 74 Ejecución prueba Elimina Estudiante Vive con

E.1.1.64. Método Gestión Estudiante Vive Con

Test run completed Results: 1/1 passed; Item(s) checked: 0			
Result	Test Name	Project	Error Message
<input type="checkbox"/>  Passed	funPsiGestionEstudianteViveConTest	EWEBPrueba	

Figura E. 75 Ejecución prueba Gestión Estudiante Vive Con

E.1.1.65. Método Gestión Ficha Estudiante

<input checked="" type="checkbox"/> Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funPsiGestionFichaEstudianteTest	EWEBPrueba	

Figura E. 76 Resultado prueba Gestión Ficha Estudiante

E.1.1.66. Método Gestión Orienta Tarea

<input checked="" type="checkbox"/> Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funPsiGestionOrientaTareaTest	EWEBPrueba	

Figura E. 77 Resultado prueba Gestión Orienta Tarea

E.1.1.67. Método Obtener Ficha Estudiante

<input checked="" type="checkbox"/> Test run completed Results: 1/1 passed; Item(s) checked: 0				
	Result	Test Name	Project	Error Message
<input type="checkbox"/>	 Passed	funPsiObtenerFichaEstudianteTest	EWEBPrueba	

Figura E. 78 Resultado prueba Obtener Ficha Estudiante

HOJA DE VIDA



INFORMACIÓN PERSONAL

APELLIDOS: ALTAMIRANO BASTIDAS
NOMBRE: OSCAR MAURICIO
CÉDULA: 171045260-6
ESTADO CIVIL: SOLTERO
NACIONALIDAD: ECUATORIANA
DOMICILIO: CIUDADELA SOLANDA, SECTOR 4, SMZ 1, MZ O, CASA 4,
CALLE JUAN BARRETO
TELÉFONO: 2 680 809 - 0984660324
FECHA NACIMIENTO: QUITO, 7 DE AGOSTO 1975

ESTUDIOS REALIZADOS

PRIMARIA

ESCUELA REPÚBLICA DE RUMANIA

SECUNDARIA

COLEGIO NACIONAL "SANTIAGO DE GUAYAQUIL"

TITULO: BACHILLER EN CIENCIAS FISICO MATEMATICAS

SUPERIOR

ESCUELA POLITÉCNICA DEL EJÉRCITO

EGRESADO FACULTAD DE INGENIERIA DE SISTEMAS E INFORMÁTICA

EXPERIENCIA LABORAL

1997-2000 SYSTEMEV SOPORTE TÉCNICO

COLEGIO MILITAR N° 10 "ABDÓN CALDERÓN": CENTRO INFORMATICA DESDE EL 2000 - 2013.

DISTRITO SEIS MINISTERIO DE EDUCACIÓN: 2013 HASTA LA PRESENTE FECHA:
DEPARTAMENTO DE LAS TIC'S

CONOCIMIENTOS ADQUIRIDOS

MANTENIMIENTO Y ENSAMBLAJE DE COMPUTADORAS

CCNA 1 CISCO

CCNA 2 CISCO

CCNA 3 CISCO

CABLEADO ESTRUCTURADO

DISEÑO DE PAGINAS WEB

TENDENCIAS INFORMATICAS DEL FUTURO

REDES Y COMUNICACIÓN

MANEJO DOCUMENTAL

ADMINISTRACION LINUX

SUFICIENCIA EN INGLES

ADMINISTRADOR QUIPUX

ADMINISTRADOR DE RED

ADMINISTRADOR DEL SISTEMA ACADÉMICO DEL COMIL10

ADMINISTRADOR DE PROYECTOS DE INNOVACIÓN PARA LA INSTITUCIÓN

SOPORTE TÉCNICO A USUARIOS

SENSIBILIZACION EN LA NORMA ISO 9001

CURSO DE CONTRATACIÓN PÚBLICA Y ADMINISTRACIÓN DE CONTRATOS

REFERENCIA

MGS. MIGUEL ROMAN: DIRECTOR DEL DISTRITO OCHO TELÉFONO: 087330951

ING.SANTIAGO NORIEGA: EMPRESA ELECTRICA QUITO TECNOLOGIAS DE LA
INFORMACION TELEF: 0997066818

OSCAR SALOMÓN CÁRATE PACHECO



PERSONAL DATA

Names	: Oscar Salomón
Last names	: Cárate Pacheco
Civil State	: Soltero
Place and Date of birth	: Quito, Mayo 25 de 1978
C.I.	: 171533668-9
Address	: Santa Rita Av. Rumichaca lote S26-112 y Palmales
Telephone	: 2845-842 / 095772776

OBJECTIVE

Obtener un desarrollo profesional y personal, basado en el respeto, trabajo, responsabilidad, realizando el trabajo de una manera eficiente, eficaz con el fin de generar valor agregado en las funciones y tareas que desempeñe.

EDUCATION

PRIMARY:

Escuela Municipal Experimental "Sucre"

SECUDARY:

Colegio Nacional Experimental "Juan Pío Montúfar"
Bachiller en Físico Matemático

UNIVERSITY:

Escuela Superior Politécnica del Ejercito E.S.P.E.
Facultad de Sistemas
Ingeniería en Sistemas de Información

LANGUAGE

Suficiencia Idioma Ingles E.S.P.E.
Proficiencia Idioma Ingles C.E.C.

EXPERIENCE

C.A. EL COMERCIO.

Cargo: Técnico Soporte.
Funciones: Help Desk.
Clientes: Áreas de la empresa.
Tiempo: Desde Junio del 2003 hasta Abril 2005.
2 años 9 meses.

Solar Solution - Empresa de Desarrollo de Software.

Cargo: Ingeniero de Desarrollo.
Funciones: Encargado de todo el proceso de arquitectura y desarrollo del ciclo de vida de la aplicación Web (Análisis, Diseño, Desarrollo, Producción, Mantenimiento).
Se realizó una reingeniería de una aplicación antigua que servía como base para la construcción del nuevo modelo.
Producto: Desarrollo de una Aplicación Web Aspx compuestas de varios módulos como: Administración, Pedido, Facturación, Producción, Contabilidad, Recursos Humanos.
Clientes: Empresas pequeñas y medianas de la ciudad de Quito.
Tiempo: Desde Abril del 2005 hasta la Septiembre del 2007.
2 años 5 meses

Seriva Inc - División Colombia - Empresa de Desarrollo de Software.

Cargo: Desarrollo.
Funciones: Utilización de Power Designer para documentación y reingeniería de los modelos lógicos y físicos de los módulos desarrollados hasta esa fecha.
Participación en el desarrollo de los módulos (Opciones, ventanilla).
Desarrollo del proceso de migración de la versión 1.2.1 a la versión 1.3 instalada en el Banco Popular de Colombia.
Creación de varios reportes para la versión 1.3.
Soporte en el Banco Popular una vez instala la nueva versión.
Productos: Módulos Financieros, Tesorería (repos, Títulos Valores, Interbancarios, opciones, Ventanilla, Limites).
Clientes: Banco Popular de Colombia, Banco de Crédito de Colombia, Banco Occidental de Colombia.
Tiempo: Septiembre del 2007 hasta Agosto 2008.
1 año

Diners Club Ecuador.

Área: Crm Analítico, ahora UGI (Unidad de Gestión de la Información).
Cargo: Oficial Senior, Especialista Senior.
Funciones: Generación de información para el área de Negocios.
Desarrollo de aplicaciones Web para el área de negocios.
Generación de reportes.
Generación de lógicas como por ejemplo círculo familiar, PipeLine.
Tiempo: Desde Agosto 2008 hasta la presente fecha.
7 años

HOJA DE LEGALIZACIÓN DE FIRMAS

ELABORADO POR

Sr. Altamirano Bastidas Oscar Mauricio

Sr. Carate Pacheco Oscar Salomón

DIRECTOR DE CARRERA

Ing. Mauricio Campaña

Sangolquí, Julio 2015