

**SIGLHA – SISTEMA DE INFORMACIÓN DE GLOBAL HUMANITARIA PARA EL  
MANEJO DEL PROGRAMA DE APADRINAMIENTO EN COLOMBIA**

**JOSE LUIS GALLON QUIÑONES**

**UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS  
ABRIL DE 2013**

**SIGLHA – SISTEMA DE INFORMACIÓN DE GLOBAL HUMANITARIA PARA EL  
MANEJO DEL PROGRAMA DE APADRINAMIENTO EN COLOMBIA**

**JOSE LUIS GALLON QUIÑONES**

**Trabajo de Grado presentado como requisito parcial para optar al título de  
Ingeniero de Sistemas**

**Ing. RUBIEL BAYARDO LEAL BERNAL  
Asesor**

**UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS  
ABRIL DE 2013**

**Las ideas y conclusiones aportadas en la tesis de grado son responsabilidad exclusiva de los autores”**

**Artículo 1º de Acuerdo N° 324 de octubre 11 de 1966 emanado del Honorable Consejo Directivo de la Universidad de Nariño.**

Nota de aceptación

---

---

---

---

---

---

---

Firma del presidente del jurado

---

Firma del presidente del jurado

---

Firma del jurado

---

Firma del presidente del jurado

San Juan de Pasto, Septiembre 2 de 2013

## **AGRADECIMIENTOS**

Primero agradecer a Dios y a mi madre quien fue y sigue siendo mi mejor maestra.

A mis profesores de primaria y bachillerato a quienes les debo haberme dado las herramientas y descubierto en mi las cualidades y aptitudes necesarias para escoger esta carrera.

A la Universidad de Nariño, por su valiosa y gran labor formativa y académica.

A mis profesores universitarios que más que profesores se comportaron como personas agradables con interés de verme superar.

A la fundación Global Humanitaria que se ganó no solo mi gratitud por haberme permitido realizarme en esa institución, sino, por haberme despertado ese sentimiento humanista del que muchos a veces carecemos.

A Lidia Paipa por sus palabras de aliento y positivismo y a Jazmín Figueroa ser la persona que terminó de darme el último empuje, sin ella no hubiese logrado culminar con éxitos este proyecto.

## DEDICATORIA

“El camino fue duro y largo, pero lo importante fue llegar, en este camino empinado tuve muchos tropiezos y caídas, pero Dios estuvo presente siempre para de una u otra manera hacer que no me rindiera, a través de personas que me entregaron su apoyo, palabras de aliento y compañía.

A ellos va dedicado este trabajo que marca mi vida y que las hace ocupar un espacio muy especial en mi corazón.

A Dios, porque sin su presencia en mi vida no podría tan siquiera haber empezado.

A mis hijos que son la razón más importante para superarme y a mi Madre que sacrificó tanto para verme alcanzar este sueño.

A mis hermanos que siempre me desean lo mejor y que al igual que mi madre deseaban verme triunfante”.

## RESUMEN

Este documento los guiará a través de las diferentes fases de planeación, desarrollo e implementación de SIGLHA: Sistema de Información de Global Humanitaria para el manejo del Programa de Apadrinamiento en Colombia, elaborado como trabajo de Grado.

**SIGLHA** es una solución informática con características de Unificación, Depuración, Actualización, Control, Sostenimiento y Manejo, en la cual se administra una base de datos centralizada de niños/as y padrinos afiliados al Programa de APADRINAMIENTO de GLOBAL HUMANITARIA en Colombia, donde se realizan directamente las tareas ya dichas, el servicio prestado mediante la web con formularios en HTML y PHP y características gráficas que permiten alimentar la base de datos e interpretar la información para hacer consultas, generar reportes e informes de esta.

Básicamente consta de cuatro (4) módulos, a saber:

- 1. MÓDULO SIGLHA ADMINISTRACION Y PERMISOS DE USUARIOS.** Este módulo registra a los usuarios involucrados en el manejo del sistema, y asigna los permisos o privilegios que estos van a tener dependiendo del cargo asignado en la Fundación, este módulo es manejado por el Administrador del sistema, y depende de su criterio el asignar privilegios a las personas involucradas, con el visto bueno de la Dirección General de la Fundación.
- 2. MÓDULO SIGLHA MANEJO Y CONTROL.** En este módulo está el manejo y control de las bases de datos de Apadrinamiento. Cuando se habla de manejo se hace referencia a todas las tareas de búsqueda de datos usando como referencia documentos y nombre de usuario, códigos de niño, etc. y por control a las tareas de modificación, inclusión y exclusión de nuevos afiliados (niños o padrinos).
- 3. MÓDULO SIGLHA DE REPORTES E INFORMES.** En este módulo se maneja todo lo relacionado a los reportes que se necesitan periódicamente de la base de datos, tales como las actividades realizadas por los técnicos.
- 4. MÓDULO SIGLHA EXPORTAR.** Este módulo se encarga de exportar información en Excel a las diferentes sedes.

## ABSTRACT

This document will guide you through the different stages of planning, development and implementation of SIGLHA: Global Information System for managing Humanitarian Sponsorship Program in Colombia prepared as dissertation.

**SIGLHA** is a software solution with features Unification, Debugging, Update, Control, Support and Management, which is managing a centralized database of children and godparents Affiliated to the SPONSORSHIP GLOBAL HUMANITARIAN program in Colombia, where they will perform the tasks directly, the service provided by the site with forms in HTML, PHP and graphic features to feed the database and interpret information requests and generate reports from it.

It basically consists of four (4) modules, namely:

- 1. SIGLHA ADMINISTRATION AND PERMITS USERS MODULE.** This module logs the users involved in the management system, and assigns permissions or privileges that these will have depending on the position assigned in the Foundation, this module is handled by the system Manager, and depends on your criteria to assign privileges to the people involved, with the approval of the Directorate General of the Foundation.
- 2. SIGLHA MANAGEMENT AND CONTROL MODULE.** This module is the management and control of databases sponsorship. When talking about management refers to all data search tasks using as reference documents and user name, child codes, etc. and by control to modification tasks, inclusion and exclusion of new members (children or sponsors).
- 3. SIGLHA REPORTS MODULE.** This module handles everything related to reports that are needed report of the activities of the technical.
- 4. EXPORT MODULE SIGLHA.** This module is responsible for exporting information in Excel to different venues.



## CONTENIDO

	<b>Página</b>
1. MARCO TEORICO.....	46
1.1 GLOBAL HUMANITARIA .....	46
1.2 LENGUAJE DE PROGRAMACIÓN PHP. ....	49
1.3 MODELO MVC .....	49
1.4 FRAMEWORK SIGLHA.....	54
2. FASE DE PLANEACION .....	64
2.1 HISTORIAS DE USUARIOS N°1: FUNCIONAMIENTO EN LA WEB.....	64
2.2 HISTORIAS DE USUARIOS N°2: USO DEL SISTEMA .....	64
2.3 HISTORIAS DE USUARIOS N°3: BASE DE DATOS.....	65
2.4 HISTORIAS DE USUARIOS N° 4: ACL.....	65
2.5 HISTORIAS DE USUARIOS N° 5: CREAR DE ROLES .....	66
2.6 HISTORIAS DE USUARIOS N° 6: EDITAR ROLE .....	66
2.7 HISTORIAS DE USUARIOS N° 7: LISTAR ROLE .....	67
2.8 HISTORIAS DE USUARIOS N° 8: CREAR PERMISOS .....	67
2.9 HISTORIAS DE USUARIOS N° 9: LISTAR PERMISOS .....	67
2.10 HISTORIAS DE USUARIOS N° 10: ASIGNAR PERMISOS A ROLE.....	68
2.11 HISTORIAS DE USUARIOS N° 11: CREAR USUARIO .....	68
2.12 HISTORIAS DE USUARIOS N° 12: EDITAR DATOS USUARIO .....	69
2.13 HISTORIAS DE USUARIOS N° 13: EDITAR PERMISOS USUARIO .....	69
2.14 HISTORIAS DE USUARIOS N° 14: LISTAR USUARIO .....	70
2.15 HISTORIAS DE USUARIOS N° 15: ELIMINAR USUARIO.....	70

2.16 HISTORIAS DE USUARIOS N° 16: CREAR COPIAS DE SEGURIDAD.....	71
2.17 HISTORIAS DE USUARIOS N° 17: CREAR UBICACIONES.....	71
2.18 HISTORIAS DE USUARIOS N° 18: EDITAR UBICACIONES .....	72
2.19 HISTORIAS DE USUARIOS N° 19: ELIMINAR UBICACIONES .....	72
2.20 HISTORIAS DE USUARIOS N° 20: LISTAR UBICACIONES.....	73
2.21 HISTORIAS DE USUARIOS N° 21: CONSULTAR UBICACIONES .....	73
2.22 HISTORIAS DE USUARIOS N° 22: AGREGAR PADRINO.....	74
2.23 HISTORIAS DE USUARIOS N° 23: EDITAR PADRINO .....	74
2.24 HISTORIAS DE USUARIOS N° 24: ELIMINAR PADRINO.....	74
2.25 HISTORIAS DE USUARIOS N° 25: LISTAR PADRINOS.....	75
2.26 HISTORIAS DE USUARIOS N° 26: CONSULTAR PADRINOS .....	75
2.27 HISTORIAS DE USUARIOS N° 27: AGREGAR INFORMACIÓN NIÑO .....	76
2.28 HISTORIAS DE USUARIOS N° 28: EDITAR INFORMACIÓN NIÑO .....	76
2.29 HISTORIAS DE USUARIOS N° 29: ELIMINAR INFORMACIÓN NIÑO .....	76
2.30 HISTORIAS DE USUARIOS N° 30: LISTAR INFORMACIÓN NIÑO .....	77
2.31 HISTORIAS DE USUARIOS N° 31: CONSULTAR INFORMACIÓN NIÑOS ..	77
2.32 HISTORIAS DE USUARIOS N° 32: CREAR APADRINAMIENTO .....	78
2.33 HISTORIAS DE USUARIOS N° 33: EDITAR APADRINAMIENTO .....	78
2.34 HISTORIAS DE USUARIOS N° 34: ELIMINAR APADRINAMIENTO.....	79
2.35 HISTORIAS DE USUARIOS N° 35: LISTAR APADRINAMIENTOS.....	79
2.36 HISTORIAS DE USUARIOS N° 36: CONSULTAR APADRINAMIENTO.....	80
2.37 HISTORIAS DE USUARIOS N° 37: AGREGAR DIBUJO.....	80
2.38 HISTORIAS DE USUARIOS N° 38: EDITAR DIBUJO .....	81
2.39 HISTORIAS DE USUARIOS N° 39: ELIMINAR DIBUJO.....	81

2.40 HISTORIAS DE USUARIOS N° 40: LISTAR DIBUJO .....	82
2.41 HISTORIAS DE USUARIOS N° 41: CONSULTAR DIBUJO.....	82
2.42 HISTORIAS DE USUARIOS N° 42: CONSULTAR ESTADO NIÑO .....	82
2.43 HISTORIAS DE USUARIOS N° 43: EXPORTAR DATOS A ESPAÑA.....	83
2.44 HISTORIAS DE USUARIOS N° 44: REPORTAR BITÁCORA DE USO .....	83
3. FASE DE DISEÑO .....	84
3.1 TARJETA CRC NO 1: NUEVO ROLE .....	84
3.2 TARJETA CRC NO 2: EDITAR ROLE.....	84
3.3 TARJETA CRC NO 3: LISTAR ROLE .....	85
3.4 TARJETA CRC NO 4: CREAR PERMISOS .....	86
3.5 TARJETA CRC NO 5: LISTAR PERMISOS .....	86
3.6 TARJETA CRC NO 6: ASIGNAR PERMISOS A ROLE .....	87
3.7 TARJETA CRC NO 7: CREAR USUARIO.....	87
3.8 TARJETA CRC NO 8: EDITAR DATOS DE USUARIO.....	88
3.9 TARJETA CRC NO 9: EDITAR PERMISOS USUARIO .....	88
3.10 TARJETA CRC NO 10: LISTAR USUARIO.....	89
3.11 TARJETA CRC NO 11: ELIMINAR USUARIO .....	89
3.12 TARJETA CRC NO 12: CREAR COPIAS DE SEGURIDAD .....	90
3.13 TARJETA CRC NO 13: CREAR UBICACIONES .....	90
3.14 TARJETA CRC NO 14: EDITAR UBICACIONES.....	91
3.15 TARJETA CRC NO 15: ELIMINAR UBICACIONES .....	91
3.16 TARJETA CRC NO 16: LISTAR UBICACIONES .....	92
3.17 TARJETA CRC NO 17: CONSULTAR UBICACIONES.....	93
3.18 TARJETA CRC NO 18: AGREGAR PADRINO .....	93

3.19	TARJETA CRC NO 19: EDITAR PADRINO .....	94
3.20	TARJETA CRC NO 20: ELIMINAR PADRINO .....	94
3.21	TARJETA CRC NO 21: LISTAR PADRINO.....	95
3.22	TARJETA CRC NO 22: CONSULTAR PADRINOS.....	95
3.23	TARJETA CRC NO 23: AGREGAR INFORMACIÓN NIÑO .....	96
3.24	TARJETA CRC NO 24: EDITAR INFORMACIÓN NIÑO .....	96
3.25	TARJETA CRC NO 25: ELIMINAR INFORMACIÓN NIÑO .....	97
3.26	TARJETA CRC NO 26: LISTAR INFORMACIÓN NIÑO.....	97
3.27	TARJETA CRC NO 27: CONSULTAR INFORMACIÓN NIÑOS.....	98
3.28	TARJETA CRC NO 28: CREAR APADRINAMIENTO.....	98
3.29	TARJETA CRC NO 29: EDITAR APADRINAMIENTO .....	99
3.30	TARJETA CRC NO 30: ELIMINAR APADRINAMIENTO .....	99
3.31	TARJETA CRC NO 31: LISTAR APADRINAMIENTO.....	100
3.32	TARJETA CRC NO 32: CONSULTAR APADRINAMIENTO .....	101
3.33	TARJETA CRC NO 33: AGREGAR DIBUJO .....	101
3.34	TARJETA CRC NO 34: EDITAR DIBUJO .....	102
3.35	TARJETA CRC NO 35: ELIMINAR DIBUJO .....	102
3.36	TARJETA CRC NO 36: LISTAR DIBUJO.....	103
3.37	TARJETA CRC NO 37: CONSULTAR DIBUJO .....	103
3.38	TARJETA CRC NO 38: EXPORTAR DATOS A ESPAÑA.....	104
4.1	UNIDAD DE PRUEBA NO1: ACLCONTROLLER .....	105
4.2	UNIDAD DE PRUEBA NO 2: REGISTROCONTROLLER.....	106
4.3	UNIDAD DE PRUEBA NO 3: USUARIOS-INDEXCONTROLLER.....	107
5.1	PRUEBAS DE ACEPTACIÓN NO.1: INICIAR LA APLICACIÓN.....	108

5.2	PRUEBAS DE ACEPTACIÓN NO. 2: INICIAR SESIÓN .....	110
5.3	PRUEBAS DE ACEPTACIÓN NO. 3: INGRESAR NUEVO USUARIO .....	112
5.4	PRUEBAS DE ACEPTACIÓN NO. 4: EDITAR USUARIO .....	115
5.5	PRUEBA DE ACEPTACIÓN NO 5: CREAR UN ROLE .....	117
5.6	PRUEBAS DE ACEPTACIÓN NO. 6: ACL.....	119
5.7	PRUEBAS DE ACEPTACIÓN NO. 7: LISTAR USUARIO .....	122
5.8	PRUEBAS DE ACEPTACIÓN NO. 8: ELIMINAR USUARIO.....	123
5.9	PRUEBAS DE ACEPTACIÓN NO. 9: CREAR COPIAS DE SEGURIDAD...	125
5.10	PRUEBAS DE ACEPTACIÓN NO. 10: CREAR UBICACIONES.....	126
5.11	PRUEBAS DE ACEPTACIÓN NO. 11: EDITAR UBICACIÓN.....	127
5.12	PRUEBAS DE ACEPTACIÓN NO. 12: LISTAR UBICACIÓN .....	129
5.13	PRUEBAS DE ACEPTACIÓN NO. 13: ELIMINAR UBICACIÓN .....	130
5.14	PRUEBAS DE ACEPTACIÓN NO. 14: AGREGAR PADRINO.....	132
5.15	PRUEBAS DE ACEPTACIÓN NO. 15: EDITAR PADRINO .....	133
5.16	PRUEBAS DE ACEPTACIÓN NO. 16: LISTAR PADRINO .....	134
5.17	PRUEBAS DE ACEPTACIÓN NO. 17: ELIMINAR PADRINO.....	135
5.18	PRUEBAS DE ACEPTACIÓN NO. 18: AGREGAR INFORMACIÓN NIÑO..	137
5.19	PRUEBAS DE ACEPTACIÓN NO. 19: EDITAR INFORMACIÓN NIÑO .....	138
5.20	PRUEBAS DE ACEPTACIÓN NO. 20: ELIMINAR AHIJADO .....	140
5.21	PRUEBAS DE ACEPTACIÓN NO. 21: LISTAR NIÑO .....	141
5.22	PRUEBAS DE ACEPTACIÓN NO. 22: CREAR APADRINAMIENTO .....	142
5.23	PRUEBAS DE ACEPTACIÓN NO. 23: EDITAR APADRINAMIENTO.....	144
5.24	PRUEBAS DE ACEPTACIÓN NO. 24: ELIMINAR APADRINAMIENTO.....	145
5.25	PRUEBAS DE ACEPTACIÓN NO. 25: LISTAR APADRINAMIENTO .....	147

5.26 PRUEBAS DE ACEPTACIÓN NO. 26: AGREGAR DIBUJO.....	148
5.27 PRUEBAS DE ACEPTACIÓN NO. 27: EDITAR DIBUJO.....	149
5.28 PRUEBAS DE ACEPTACIÓN NO. 28: ELIMINAR DIBUJO.....	150
5.29 PRUEBAS DE ACEPTACIÓN NO. 29: LISTAR DIBUJO .....	152
5.30 PRUEBAS DE ACEPTACIÓN NO. 030: EXPORTAR DATOS A ESPAÑA ..	153
5.31 PRUEBAS DE ACEPTACIÓN NO. 031: REPORTAR BITÁCORA DE USO	155
6. DISEÑO DE DATOS .....	157
7. MODELO DE CLASES.....	161
8. DIAGRAMA DE PAQUETES.....	162
9. DIAGRAMA DE CLASES .....	168
10. LISTADO DE CLASES.....	169

## LISTA DE TABLAS

### Página

Tabla 1: Historias de Usuarios N°1: Funcionamiento en la Web.....	64
Tabla 2: Historias de Usuarios N°2: Uso del Sistema .....	64
Tabla 3: Historias de Usuarios N°3: Base de Datos.....	65
Tabla 4: Historias de Usuarios N° 4: Listas de Control de Acceso – ACL.....	65
Tabla 5: Historias de Usuarios N° 5: Crear de Roles .....	66
Tabla 6: Historias de Usuarios N° 6: Editar Role .....	66
Tabla 7: Historias de Usuarios N° 7: Listar Role .....	67
Tabla 8: Historias de Usuarios N° 8: Crear Permisos .....	67
Tabla 9: Historias de Usuarios N° 9: Listar Permisos .....	67
Tabla 10: Historias de Usuarios N° 10: Asignar permisos a Role .....	68
Tabla 11: Historias de Usuarios N° 11: Crear Usuario .....	68
Tabla 12: Historias de Usuarios N° 12: Editar Datos Usuario .....	69
Tabla 13: Historias de Usuarios N° 13: Editar Permisos Usuario.....	69
Tabla 14: Historias de Usuarios N° 14: Listar Usuario .....	70
Tabla 15: Historias de Usuarios N° 15: Eliminar Usuario .....	70
Tabla 16: Historias de Usuarios N° 16: Crear copias de seguridad .....	71
Tabla 17: Historias de Usuarios N° 17: Crear Ubicaciones.....	71
Tabla 18: Historias de Usuarios N° 18: Editar Ubicaciones .....	72
Tabla 19: Historias de Usuarios N° 19: Eliminar Ubicaciones.....	72
Tabla 20: Historias de Usuarios N° 20: Listar Ubicaciones.....	73

Tabla 21: Historias de Usuarios N° 21: Consultar Ubicaciones .....	73
Tabla 22: Historias de Usuarios N° 22: Agregar Padrino .....	74
Tabla 23: Historias de Usuarios N° 23: Editar Padrino .....	74
Tabla 24: Historias de Usuarios N° 24: Eliminar Padrino .....	74
Tabla 25: Historias de Usuarios N° 25: Listar Padrinos .....	75
Tabla 26: Historias de Usuarios N° 26: Consultar Padrinos.....	75
Tabla 27: Historias de Usuarios N° 27: Agregar Información Niño .....	76
Tabla 28: Historias de Usuarios N° 28: Editar Información Niño.....	76
Tabla 29: Historias de Usuarios N° 29: Eliminar Información Niño .....	76
Tabla 30: Historias de Usuarios N° 30: Listar Información Niño .....	77
Tabla 31: Historias de Usuarios N° 31: Consultar Información Niños .....	77
Tabla 32: Historias de Usuarios N° 32: Crear Apadrinamiento .....	78
Tabla 33: Historias de Usuarios N° 33: Editar Apadrinamiento.....	78
Tabla 34: Historias de Usuarios N° 34: Eliminar Apadrinamiento .....	79
Tabla 35: Historias de Usuarios N° 35: Listar Apadrinamientos.....	79
Tabla 36: Historias de Usuarios N° 36: Consultar Apadrinamiento.....	80
Tabla 37: Historias de Usuarios N° 37: Agregar Dibujo .....	80
Tabla 38: Historias de Usuarios N° 38: Editar Dibujo.....	81
Tabla 39: Historias de Usuarios N° 39: Eliminar Dibujo .....	81
Tabla 40: Historias de Usuarios N° 40: Listar Dibujo .....	82
Tabla 41: Historias de Usuarios N° 41: Consultar Dibujo.....	82
Tabla 42: Historias de Usuarios N° 42: Consultar Estado Niño .....	82
Tabla 43: Historias de Usuarios N° 43: Exportar Datos a España .....	83
Tabla 44: Historias de Usuarios N° 44: Reportar Bitácora de Uso de SIGLHA.....	83



Tabla 45: Tarjeta CRC No 1: Nuevo Role.....	84
Tabla 46: Tarjeta CRC No 2: Editar Role.....	84
Tabla 47: Tarjeta CRC No 3: Listar Role .....	85
Tabla 48: Tarjeta CRC No 4: Crear Permisos.....	86
Tabla 49: Tarjeta CRC No 5: Listar Permisos.....	86
Tabla 50: Tarjeta CRC No 6: Asignar permisos a Role.....	87
Tabla 51: Tarjeta CRC No 7: Crear Usuario .....	87
Tabla 52: Tarjeta CRC No 8: Editar Datos de Usuario.....	88
Tabla 53: Tarjeta CRC No 9: Editar Permisos Usuario .....	88
Tabla 54: Tarjeta CRC No 10: Listar Usuario.....	89
Tabla 55: Tarjeta CRC No 11: Eliminar Usuario .....	89
Tabla 56: Tarjeta CRC No 12: Crear Copias de Seguridad .....	90
Tabla 57: Tarjeta CRC No 13: Crear Ubicaciones .....	90
Tabla 58: Tarjeta CRC No 14: Editar Ubicaciones.....	91
Tabla 59: Tarjeta CRC No 15: Eliminar Ubicaciones .....	91
Tabla 60: Tarjeta CRC No 16: Listar Ubicaciones .....	92
Tabla 61: Tarjeta CRC No 17: Consultar Ubicaciones.....	93
Tabla 62: Tarjeta CRC No 18: Agregar Padrino.....	93
Tabla 63: Tarjeta CRC No 19: Editar Padrino .....	94
Tabla 64: Tarjeta CRC No 20: Eliminar Padrino .....	94
Tabla65 Tarjeta CRC No 21: Listar Padrino.....	95
Tabla67 Tarjeta CRC No 22: Consultar Padrinos .....	95
Tabla68 Tarjeta CRC No 23: Agregar Información Niño .....	96
Tabla69 Tarjeta CRC No 24: Editar Información Niño .....	96

Tabla70 Tarjeta CRC No 25: Eliminar Información Niño.....	97
Tabla71 Tarjeta CRC No 26: Listar Información Niño.....	97
Tabla72 Tarjeta CRC No 27: Consultar Información Niños .....	98
Tabla73 Tarjeta CRC No 28: Crear Apadrinamiento.....	98
Tabla74 Tarjeta CRC No 29: Editar Apadrinamiento .....	99
Tabla75 Tarjeta CRC No 30: Eliminar Apadrinamiento.....	99
Tabla76 Tarjeta CRC No 31: Listar Apadrinamiento.....	100
Tabla77 Tarjeta CRC No 32: Consultar Apadrinamiento .....	101
Tabla78 Tarjeta CRC No 33: Agregar Dibujo.....	101
Tabla79 Tarjeta CRC No 34: Editar Dibujo .....	102
Tabla80 Tarjeta CRC No 35: Eliminar Dibujo.....	102
Tabla81 Tarjeta CRC No 36: Listar Dibujo.....	103
Tabla82 Tarjeta CRC No 37: Consultar Dibujo .....	103
Tabla83 Tarjeta CRC No 38: Exportar Datos a España.....	104
Tabla84 Unidad de prueba No1: ACLController.....	105
Tabla85 Unidad de prueba No 2: RegistroController .....	106
Tabla86 Unidad de prueba No 3: Usuarios-indexController.....	107
Tabla 87 Pruebas de aceptación No.1: Iniciar la aplicación.....	108
Tabla88 Pruebas de aceptación No. 2: Iniciar sesión.....	110
Tabla89 Pruebas de aceptación No. 3: Ingresar Nuevo Usuario .....	112
Tabla90 Pruebas de aceptación No. 4: Editar Usuario .....	115
Tabla91 Prueba de Aceptación No 5: Crear un Role .....	117
Tabla 92: Pruebas de aceptación No. 6: Lista de Control de Acceso ACL .....	119
Tabla 93: Pruebas de aceptación No. 7: Listar Usuario .....	122

Tabla 94: Pruebas de aceptación No. 8: Eliminar Usuario.....	123
Tabla 95: Pruebas de aceptación No. 9: Crear Copias de Seguridad.....	125
Tabla 96: Pruebas de aceptación No. 10: Crear Ubicaciones .....	126
Tabla 97: Pruebas de aceptación No. 11: Editar Ubicación .....	127
Tabla 98: Pruebas de aceptación No. 12: Listar Ubicación .....	129
Tabla 99: Pruebas de aceptación No. 13: Eliminar Ubicación .....	130
Tabla 100: Pruebas de aceptación No. 14: Agregar Padrino .....	132
Tabla 101: Pruebas de aceptación No. 15: Editar padrino.....	133
Tabla 102: Pruebas de aceptación No. 16: Listar padrino .....	134
Tabla 103: Pruebas de aceptación No. 17: Eliminar Padrino.....	135
Tabla 104: Pruebas de aceptación No. 18: Agregar Información Niño .....	137
Tabla 105: Pruebas de aceptación No. 19: Editar Información Niño.....	138
Tabla 106: Pruebas de aceptación No. 20: Eliminar ahijado .....	140
Tabla 107: Pruebas de aceptación No. 21: Listar niño .....	141
Tabla 108: Pruebas de aceptación No. 22: Crear Apadrinamiento .....	142
Tabla 109: Pruebas de aceptación No. 23: Editar Apadrinamiento.....	144
Tabla 110: Pruebas de aceptación No. 24: Eliminar apadrinamiento .....	145
Tabla 111: Pruebas de aceptación No. 25: Listar apadrinamiento.....	147
Tabla 111: Pruebas de aceptación No. 26: Agregar Dibujo .....	148
Tabla 112: Pruebas de aceptación No. 27: Editar Dibujo .....	149
Tabla 113: Pruebas de aceptación No. 28: Eliminar Dibujo.....	150
Tabla 114: Pruebas de aceptación No. 29: Listar Dibujo .....	152
Tabla 115: Pruebas de aceptación No. 030: Exportar Datos a España .....	153
Tabla 116: Pruebas de aceptación No. 031: Reportar Bitácora de Uso.....	155

## LISTA DE FIGURAS

	<b>Página</b>
Figura 1: Áreas de Trabajo de Global Humanitaria.....	47
Figura 2: Relación entre módulos del patrón MVC .....	51
Figura 3: MODELO MVC SIGLHA– (MODELO-CONTROLADOR-VISTA).....	53
Figura 4: Sistema de archivos FRAMEWORK.....	55
Figura 5: Entidad- Relación .....	157
Figura 6: Modelo de Clases. ....	161
Figura 7: Diagramas de paquetes.....	162
Figura 8: FRAMEWORK .....	162
Figura 9: Administración/Usuario .....	163
Figura 10: Usuarios.....	164
Figura 11: Manejo y control .....	165
Figura 12: Crios o Ahijados.....	165
Figura 13: Ubicación lugares .....	166
Figura 14 Reportes e Informes .....	166
Figura 15: Exportar .....	167
Figura 16: Diagrama de clases para el patrón MVC. ....	168
Figura 17: Clase Controlador principal del FRAMEWORK. ....	169
Figura 18: Clase ERROR del FRAMEWORK. ....	169

## GLOSARIO

<b>Access:</b>	<p>Microsoft Access es una herramienta de Microsoft para la definición y manipulación de bases de datos.</p> <p>Por sus características y rendimiento como el programa más utilizado para el manejo de bases de datos en el mercado de la computación y la informática, la presentación en pantalla de los menús, barras de herramientas, botones, etc. y demás elementos que lo constituyen, sí han tenido cambios significativos, pero siempre enfocados hacia el mayor rendimiento, desempeño y facilidad en su manejo por quienes lo utilizan para realizar tareas diversas en cualquier ambiente, sea esté de trabajo o escolar.<sup>1</sup></p>
<b>Ahijado:</b>	Persona que es apoyada o protegida, en relación con la que la apoya o protege
<b>Apadrinamiento</b>	Acción de apadrinar
<b>Apadrinar</b>	Protección o favor que una persona o entidad da a otra para hacer algo
<b>Aplicación</b>	Programa informático que realiza una función determinada
<b>Auxiliar</b>	Que auxilia o sirve de ayuda
<b>Base de Datos</b>	Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.
<b>Campaña de dibujo</b>	Programa anual donde los niños realizan un dibujo que es enviado a su padrino correspondiente.
<b>Donativos</b>	Cantidad de dinero o conjunto de medicinas, alimentos, ropas u otros objetos que se da voluntariamente a una persona o grupo para ayudar a cubrir sus necesidades

---

<sup>1</sup> Manual BasicoAcces 2007. [www.proulex.com](http://www.proulex.com). <http://www.proulex.com/computo/oe/doc/basico-access2007.pdf> consultado el día 30 de Noviembre de 2013. s.f. (último acceso: 30 de Noviembre de 2013).

<b>Framework</b>	Un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.
<b>Hardware</b>	se refiere a todas las partes tangibles de un sistema informático; sus componentes son: eléctricos, electrónicos, electromecánicos y mecánicos
<b>Informe</b>	es un trabajo escrito, cuyos resultados o cuyo producto es esperado por personas distintas a quien lo realiza, o bien el mismo es encargado por terceros
<b>Institución Educativa</b>	Centro de enseñanza
<b>Interfaz</b>	Es el lugar de la interacción, el espacio donde se desarrollan los intercambios y sus manualidades.
<b>Núcleo familiar</b>	Se refiere a familiares o personas con la que se comparte un vínculo afectivo
<b>Padrino</b>	Persona que apadrina
<b>Permiso</b>	Autorización para hacer algo que estaba prohibido y es otorgado por alguien de autoridad
<b>Proyectos</b>	Idea de una cosa que se piensa hacer y para la cual se establece un modo determinado y un conjunto de medios necesarios
<b>Reporte</b>	Informe o noticia del estado actual de algo
<b>Responsabilidad social empresarial</b>	Se define como la contribución activa y voluntaria al mejoramiento social, económico y ambiental por parte de las empresas, generalmente con el objetivo de mejorar su situación competitiva, valorativa y su valor añadido. El sistema de evaluación de desempeño conjunto de la organización en estas áreas es conocido como el triple resultado.
<b>Role</b>	Función, papel de un usuario en el sistema
<b>Sede educativa</b>	Centro educativo hace parte de una institución educativa
<b>SIGLHA</b>	Sistema de Información para el Programa de Apadrinamiento de Global Humanitaria – Colombia
<b>Sistema</b>	(System). Un sistema es un conjunto de partes o elementos organizados y relacionados que

	interactúan entre sí para lograr un objetivo. Los sistemas reciben (entrada) datos, energía o materia del ambiente y proveen (salida) información, energía o materia.
<b>Socio</b>	Persona que pertenece a un grupo con un fin determinado
<b>Software</b>	Conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware.
<b>Técnico</b>	Persona que posee conocimientos o habilidades especializadas en relación con una ciencia o una actividad determinada
<b>Ubicación</b>	Situación o lugar en el que se encuentra alguien o algo
<b>Usuario</b>	Se aplica a la persona que usa habitualmente algo.
<b>Zona</b>	Parte delimitada de cualquier superficie

## INTRODUCCIÓN

La fundación Global Humanitaria, es una Organización sin ánimo de lucro que tiene como eje central en su quehacer el proceso de apadrinamiento; dicho proceso cuenta con la interacción de diferentes actores que realizan desde la recolección hasta la sistematización de información.

El sistema de información manejado por la institución se basa en el programa Excel apoyado a través de documentación en físico que viabiliza o dificulta la sistematización de la información, con periodos de tiempos superiores a una semana para la realización de la actualización de la misma. En un programa tan activo y dinámico como el de apadrinamiento, el no contar con un sistema de información en línea que permita desde la creación de roles y usuarios la interacción de diversos actores con la información, se impide el seguimiento adecuado a los ahijados y un reporte a los padrinos; pero principalmente, esto no permite el flujo adecuado de la información y con ello la puesta en marcha de acciones correctivas ante las problemáticas evidenciadas.

Partiendo de lo expuesto en el párrafo anterior, surge SIGLHA como un sistema de información para el programa de apadrinamiento que permite unificar y mantener actualizada en tiempo real la base de datos para las sedes de Tumaco, Bogotá y España, permitiendo la interacción de diversos usuarios con diferente roles en un mismo momento, sin que esto traiga como consecuencia pérdida, retraso o alteración de la información.

A lo largo de este documento, se expondrá el proceso llevado a cabo para la realización del sistema de información con cuatro módulos, el cual parte de la creación de un FRAMEWORK propio que sirve como plataforma para la programación del sistema, todo basado en el modelo MVC.





## DESCRIPCIÓN DEL PROBLEMA

### PLANTEAMIENTO DEL PROBLEMA

El sistema de información del programa de apadrinamiento, no contaba con un sistema óptimo, que se retroalimente y generara consultas e informes de manera automática, puesto que esta información se llevaba mediante hojas de cálculo manejadas a través Microsoft Office Excel 2007 o Microsoft Access 2003, lo cual resultaba siendo engorroso y poco eficiente, debido a que no permitía la actualización en tiempo real, de los datos registrados por los técnicos de campo, que son los encargados de alimentar la base de datos y cuya labor se realizaba de manera manual.

Por otra parte, la descentralización de la base de datos de Apadrinamiento no permitía mantener actualizada la información de forma oportuna, debido a los procesos de cambios realizados en cada sede, puesto que diariamente se presentan continuos movimientos por parte de los técnicos de campo, situación nada favorable para la Fundación o en su defecto para un Padrino que desee conocer el estado actual de su ahijado.

Adicionalmente a esto, cabe mencionar que en la sede Tumaco se manejaba un periodo de 15 días para llevar a cabo esta actualización de datos. Información que luego es enviada a la sede Bogotá, quien a su vez actualiza la información tanto de padrinos Colombianos, como de beneficiarios y, consecutivamente envía la base de datos a España, en donde se actualiza la información de sus Padrinos, y la reasignación de beneficiarios, para posteriormente devolver la base de datos a Colombia – Bogotá encargada de esta actualización y envió de información a la sede Tumaco, para que estos datos también sean nuevamente actualizados, esto en un periodo de un mes, situación que evidenció la carencia de tiempo, para la actualización de las tres bases de datos en las distintas sedes.

Situación que dificultaba el cumplimiento y efectividad de las acciones a realizar por medio del programa, en pro del bienestar de la población infantil afiliada y que reduce la ampliación o cobertura de los posibles beneficiarios, debido a que este trabajo, se centraba más en la actualización de las bases de datos manejadas.

## **FORMULACIÓN DEL PROBLEMA**

¿Cómo mantener la base de datos del programa de apadrinamiento de la Fundación Global Humanitaria Colombia actualizada en las sedes involucradas, para que las adiciones, modificaciones, consultas y reportes sean en tiempo real, con el fin de poder mejorar el servicio y mantener un control en la información?

## **SISTEMATIZACIÓN DEL PROBLEMA**

¿De qué manera se puede obtener una base de datos centralizada donde las diferentes sedes puedan estar sincronizadas con los cambios en la información en el Programa de Apadrinamiento de la Fundación Global Humanitaria?

¿Cómo mantener segura la información de la base de datos online para evitar pérdidas por ataques o fallos en el servidor web?

¿Cómo monitorear los procesos que realizan los usuarios en el sistema para controlar y evaluar el manejo de la información?

¿Cómo lograr que las sedes de Bogotá y España realicen las actualizaciones en la base de datos online de manera individual y masiva por medio de archivos planos?

¿Cómo ayudar a la coordinación en realizar estudios socioeconómicos de las familias de los beneficiarios del programa de apadrinamiento para la toma de decisiones y servir de apoyo a los diferentes proyectos que realiza la fundación para ubicar, seleccionar y ayudar a las familias de los beneficiarios del programa de apadrinamiento?

¿Cómo suministrar información de la base de datos de Colombia a España para ayudar al programa de apadrinamiento Español?

¿Cómo realizar las diferentes actividades o procesos del Programa de apadrinamiento en SIGLHA?

## ALCANCE Y DELIMITACIÓN

**SIGLHA** es una solución informática con características de Unificación, Actualización, Control, Sostenimiento y Manejo, en la cual se administra una base de datos centralizada de beneficiarios (niños y niñas ahijados) y clientes (padrinos) afiliados al Programa de APADRINAMIENTO de GLOBAL HUMANITARIA en Colombia, donde se realizan directamente las tareas ya dichas, el servicio prestado mediante la web con formularios en HTML y PHP y características gráficas que permiten alimentar la base de datos e interpretar la información para hacer consultas, generar reportes e informes de esta.

SIGLHA maneja información de interés y necesidad para el funcionamiento del sistema de información de las sedes de Bogotá y Barcelona España como son: los datos de los beneficiarios, de clientes y las actividades propias del proceso de apadrinamiento como son:

- Campañas de Afiliación.- proceso en el cual los técnicos recolectan datos de nuevos niños para ser ingresados al programa de apadrinamiento.
- Campaña de Entrega de Kid Escolar.- Este proceso aprovecha la entrega de kid escolares a los niños beneficiarios del programa de apadrinamiento para actualizar los datos y fotos y se realiza 2 veces al año.
- Campaña de Entrega de Dibujo: En este proceso los niños realizan un dibujo que posteriormente es entregado a los padrinos y en el cual también se aprovecha para actualizar los datos de los niños y fotos.
- Entrega de donativos: en este proceso se le entrega a los niños beneficiarios el donativo que el padrino le envía directamente a su ahijado o que alguna entidad done a la fundación y en el cual se le devuelve al padrino o entidad una carta de agradecimiento elaborada por el ahijado o los niños beneficiados con el donativo.
- Campaña de actualización: en este proceso los técnicos ubican a los niños para actualizar los datos directamente en el colegio donde se encuentren matriculados

Con la elaboración de esta nueva herramienta se logra crear un sistema que mantiene actualizada la base de datos del Programa de Apadrinamiento de Global Humanitaria en Colombia para que las sedes tengan acceso en tiempo real a las actualizaciones, modificación, afiliaciones y egresos de beneficiarios, así como

permitir la conexión de la información de afiliados beneficiados con otros programas y proyectos de la fundación, para afianzar más la relación padrino – ahijado.

Básicamente existen cuatro (4) escenarios a ser utilizados en el proyecto:

## **MÓDULO SIGLHA ADMINISTRACIÓN Y PERMISOS DE USUARIOS**

Este módulo crea copias de seguridad hechas a la base de datos y genera reportes de bitácora de uso en SIGLHA.

Además, registra a los usuarios involucrados en el manejo del sistema, y asigna los permisos o privilegios por medio de la creación roles y permisos que estos van a tener dependiendo del cargo asignado en la fundación, este módulo es manejado por el administrador del sistema, y depende de su criterio el asignar privilegios a las personas involucradas, con el visto bueno de la dirección general de la fundación.

## **MÓDULO SIGLHA MANEJO Y CONTROL**

En este módulo está el manejo y control de las bases de datos de Apadrinamiento. Cuando se habla de manejo se hace referencia a todas las tareas de búsqueda de datos usando como referencia documentos y nombre de usuario, códigos de niño, etc. y por control a las tareas de modificación, inclusión y exclusión de nuevos afiliados (niños o padrinos).

La información de los padrinos españoles y colombianos que debe ser compartida entre las distintas sedes para el control de asignación de padrinos, estado de los niños apadrinados y no apadrinados, niños dados de baja, ingreso de niños, padrinos dados de baja, ingreso de nuevos padrinos, cambio de ahijado a los padrinos, y todos los procesos que tengan relación padrino ahijado son alimentados por las sedes de Bogotá y Barcelona España quienes podrán actualizar la base de datos de SIGLHA de manera manual (uno a uno) o de manera masiva (cargue de archivo plano)

Al igual que los procesos realizados por los técnicos en las diferentes campañas ya mencionadas.

### **MÓDULO SIGLHA DE REPORTE E INFORMES**

En este módulo se maneja todo lo relacionado a los reportes que se necesitan periódicamente de la base de datos, tales como reporte de novedades aplicadas en el mes, reporte de retiros, informe de las actividades realizadas por los técnicos, historial de proyectos que los niños han sido beneficiados, estadísticas de la información e historiales tanto de beneficiarios como clientes.

### **MÓDULO SIGLHA EXPORTAR BASE DE DATOS**

Este módulo se encarga de exportar información en Excel a las diferentes sedes.

## ANTECEDENTES

El sistema de información en la Fundación Global Humanitaria llamado **APADRINAMIENTO2011** abarcaba todos los módulos anteriormente mencionados y se usaba en la ciudad de Bogotá, fue desarrollado por Global Humanitaria. Se implementó en un servidor con SQL server 2003 y una base de datos en Microsoft Office ACCES 2003 con las interfaces en macros de Visual Basic. Existen aplicaciones offline para la sede de Tumaco en cuyas aplicaciones toca exportar los datos y actualizar una vez se tenga un consolidado en Bogotá.

En la universidad de Nariño se han desarrollado los siguientes proyectos de grado de sistemas de información que aportan fundamentos para el desarrollo de SIGLHA.

- Sistema de información y documentación en la sección de talento humano EMPOPASTO S.A.E.S.P 2004. Elaborada por la estudiante de ingeniería de sistemas MARIA FERNANDA CALVACHE BENAVIDES, cuyo objetivo era lograr la efectividad y eficiencia de los procesos y procedimientos de la sección de talento humano de la empresa EMPOPASTO S.A.E.S.P en la sección de talento humano, la cual al culminar el proyecto cumplió con las expectativas y objetivos propuestos dado que el proceso de organización, mantenimiento, control, seguimiento de la documentación e información de la sección de talento humano surtió el efecto en la gestión y eficiencia del proceso normal en la realización de todas y cada una de las necesidades.
- Sistema de información en un ambiente web para la administración y control de la información que relaciona los entes secretaría de educación municipal e instituciones educativas de Pasto, Elaborado por los estudiantes de ingeniería de sistemas MARIBEL GARZÓN URBANO, MARTHA ISABEL ORTIZ VELA Y LAURA INÉS URBANO CALVACHE en 2003, cuyo objetivo era diseñar un sistema de información en un ambiente web tanto para la secretaria de educación municipal como para las instituciones educativas que les permitiera además de administrar su propia información, el flujo de información que los relaciona, supliendo así la no existencia de canales de comunicación directos que garanticen un manejo eficiente y oportuno de la misma.

El sistema de información permite llevar un mejor control de la información que circula entra la secretaria municipal de educación y las instituciones educativas, permitiendo así que a información esté disponible en el momento oportuno.

- Sistema de información bibliotecario liceo de la universidad de Nariño, elaborado por los estudiantes de ingeniería de sistemas JULIÁN ANDRÉS

MONTES SOTO Y EDMUNDO JAVIER ROSERO MONTÚFAR en 2008, cuyo objetivo era implementar un sistema de información para la gestión de la biblioteca del liceo de la universidad de Nariño.

SIBLU como fue bautizado este sistema de información permite llevar por medio de herramientas computacionales el control de los registros del material bibliográfico, los procesos de préstamos y devoluciones cuyos procesos mejoran la administración, organización y la prestación del servicio a la comunidad educativa.



## JUSTIFICACIÓN

Teniendo en cuenta, que el programa de Apadrinamiento que se desarrolla a través de la fundación Global Humanitaria, beneficia a una gran cantidad de niños, y que esta labor resulta siendo muy dispendiosa en cuanto al papeleo y los requerimientos necesarios para llevar un control de los datos manejados por las tres sedes encargadas de tramitar la información, se hace necesario la integración de un sistema de información que agilice este proceso, para lo cual lo que se propone es la implementación de SIGLHA (Sistema de Información de Global Humanitaria para el Manejo del Programa de Apadrinamiento en Colombia) al servidor web de la Fundación cuya función se basa en la actualización, control y administración de la información y procesos manejados por el programa de Apadrinamiento.

Esto con el fin de mejorar notoriamente las condiciones actuales del proceso que se viene llevando a cabo, puesto que lograría reducir algunos trámites necesarios para la entrega de información tanto a las sedes como a los padrinos.

Las solicitudes por parte de los padrinos se optimizarían permitiendo la entrega de la información inmediata y actualizada de los ahijados beneficiados por el programa al no tener que esperar que las solicitudes se transfieran entre las sedes hasta obtener la información solicitada por el padrino.

La coordinación administrativa del área realizaría un mejor ejercicio de supervisión respecto al trabajo de los técnicos de campo, encargados del desarrollo de las actividades de recolección y actualización de información, donde se pretende que estos realicen su labor de forma planificada y organizada

Pudiendo así lograr el cumplimiento de las metas mensuales establecidas por el programa cuyo objetivo central se enfoca en la realización de acciones en pro del bienestar de la población infantil afiliada y de sus familias.

## **OBJETIVOS**

### **GENERAL**

Centralizar, controlar y administrar de manera eficiente la Base de Datos del Programa de Apadrinamiento de la Fundación Global Humanitaria en Colombia mediante el desarrollo e implementación de SIGLHA en el servidor web de la Fundación.

### **ESPECÍFICOS**

- Implementar la base de datos online en el servidor web de global humanitaria para el manejo de la información de niños y padrinos del programa de Apadrinamiento que permita a las distintas sedes actualizar, consultar y descargar la información de la base de datos.
- Implementar procesos de seguridad en la Base de datos de SIGLHA para generar copias de seguridad periódicas y restauración de las mismas.
- Implementar un sistema de control de acceso y bitácora de uso en SIGLHA para controlar y evaluar las actividades de los usuarios en el sistema de información.
- Implementar proceso de exportación de la base de datos de SIGLHA en hojas de cálculo para la sede española para que alimente su sistema de información.
- Implementar los diferentes módulos de actualización, inserción de nuevos datos, eliminación, consulta y reportes a los procesos de Apadrinamiento como son el de datos de beneficiarios, datos familiares, fotos, padrinos, correspondencia, y las diferentes campañas realizadas por la fundación como son: Dibujo, Kid Escolar, Donativos y Afiliación
- Implementar proceso de reporte de actividades realizadas en SIGLHA

## 1. MARCO TEORICO

### 1.1 GLOBAL HUMANITARIA

Global Humanitaria es una organización independiente, laica y plural que mediante el apadrinamiento y la cooperación internacional, trabaja para favorecer procesos participativos de desarrollo, prestando especial atención a la infancia, a la juventud y a la mujer para garantizar la igualdad de oportunidades de las poblaciones más vulnerables y excluidas en Colombia.

La política de trabajo de la fundación está basada en el “mejoramiento continuo”, por medio del cual, se busca optimizar los recursos y consolidación como una organización seria, transparente y sólida y en el “establecimiento de alianzas estratégicas “que le permitan crecer y ampliar le gestión, a fin de ofrecer mayor cobertura y apoyar de manera responsable las regiones en las cuales interviene.

La organización trabaja en el Departamento de Nariño específicamente en los Municipios de Tumaco, Francisco Pizarro y Mosquera, y desde el año 2007 en Altos de Casuca en El Municipio de Soacha Cundinamarca, siendo el principal beneficiario de los proyectos la población infantil.

#### **Principios.**

- Respeto
- Humildad
- Coherencia
- Compromiso
- Transparencia
- Empatía
- Pragmatismo

**Áreas de trabajo de Fundación Global Humanitaria.** La fundación global humanitaria para el desarrollo de sus procesos y actividades, ha establecido una serie de áreas de trabajo que le permiten cubrir los diferentes ámbitos de participación; a continuación mencionaremos cada una de estas áreas de trabajo: *(ver figura 1)*

- Desarrollo
- Educación para el desarrollo

- Acción humanitaria
- Sectores de cooperación



**Figura 1: Áreas de trabajo de Global Humanitaria.<sup>2</sup>**

**Desarrollo sostenible.** Con el convencimiento de que la pobreza puede ser erradicada si se atacan las causas estructurales que la ocasionan, trabajamos para incidir en estas causas apoyando a las comunidades a fortalecer sus capacidades locales siendo partícipes, protagonistas y motores del desarrollo.

**Educación para el desarrollo.** Se quiere apoyar y fomentar un proceso educativo dinámico, participativo y creativo para generar conciencias críticas y constructivas

<sup>2</sup> Fundación Global Humanitaria Colombia. «[www.globalhumanitariacolombia.org](http://www.globalhumanitariacolombia.org).» s.f. [http://www.globalhumanitariacolombia.org/que\\_hace\\_global\\_humanitaria\\_en\\_colombia](http://www.globalhumanitariacolombia.org/que_hace_global_humanitaria_en_colombia) (último acceso: 1 de Abril de 2013).

que hagan a cada persona responsable, con el fin de construir una sociedad comprometida y equitativa.

**Acción humanitaria.** Implica la atención en situación de crisis humanitarias y desastres naturales con un enfoque integral teniendo en cuenta la gestión de riesgos en todas sus fases (Prevención, ayuda de emergencia y rehabilitación).

Programa de apadrinamiento. El programa de Apadrinamiento de la Fundación Global Humanitaria consiste en realizar acciones que les permite afianzar la relación Padrino- Ahijado, por medio de la sistematización de la información, atención al Socio y realización de campañas ordinarias y extraordinarias, que trasciendan en beneficio de la población infantil, sus familias y la comunidad educativa del municipio de Tumaco.

El apadrinamiento de la Fundación Global Humanitaria es una forma de contribución de personas naturales o jurídicas que le permite a la Fundación desarrollar acciones como: financiar la construcción y mantenimiento de comedores escolares; asegurar la nutrición de más de diez mil beneficiarios del municipio; adelantar proyectos en torno a la salud y el medio ambiente; capacitar y formar a jóvenes y adultos, para otorgarles las herramientas para construir unidades de negocio, y empoderamiento para que puedan forjar su futuro y cambiar sus condiciones de familias vulnerables.

Para garantizarles a los padrinos que sus ahijados están recibiendo la ayuda de sus aportes, Global Humanitaria establece medios de comunicación (correos, llamadas, visitas personales) y trata de mantener actualizada la base de datos de los ahijados (foto, edad, ubicación, grado escolar, tallas, etc.) Con el fin de estrechar entre ellos lazos que les permitan formar una relación afectiva.

El manejo del flujo de información se da entre las 3 sedes involucradas que son:

- **Tumaco:** Donde se maneja la información referente a los ahijados o beneficiarios del programa apadrinamiento. Esta información debe ser actualizada a diario por diferentes medios y enviadas en un consolidado a la sede de Bogotá.
- **Bogotá:** Esta sede cuenta con un sistema de información que maneja todo el programa de apadrinamiento; se encarga de los procesos referentes a los padrinos colombianos y los procesos de ahijados tanto colombianos como españoles. Este sistema se mantiene actualizado con la información que suministra la sede de Tumaco con la información de los beneficiarios y España quien suministra información de los padrinos Españoles.

- **España:** Es la sede principal a nivel mundial y cuenta con un sistema de información para manejar apadrinamiento a nivel mundial pues no solo trabaja con Colombia; maneja los procesos de padrinos-ahijados Españoles cuya información es suministrada por Colombia.

## 1.2 LENGUAJE DE PROGRAMACIÓN PHP.

PHP<sup>3</sup> es un lenguaje de programación interpretado, diseñado para la creación de páginas Web dinámicas. Se usa para la interpretación del lado del servidor (*server-side scripting*) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas con licencia *open source*. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios Web y en un millón de servidores, el número de sitios en PHP ha compartido algo de su preponderante dominio con otros nuevos lenguajes no tan poderosos desde agosto de 2005.

El diseño de PHP está orientado a facilitar la creación de sitios Webs, es posible crear aplicaciones con una interfaz gráfica para el usuario, utilizando diversos *frameworks* que existen en internet como extjs.

Cuando el cliente hace una petición al servidor para que le envíe una página Web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica.

## 1.3 MODELO MVC

Durante la década de los setenta, SmallTalk y algunos otros lenguajes como simula I, fueron construyendo gradualmente el paradigma de la programación orientada a objetos y estableciendo conceptos tales como objetos, clases, encapsulación, herencia y polimorfismo. Si bien dichos conceptos no están vigentes actualmente, los conceptos que dejaron en el mundo del desarrollo de software están vigentes en la actualidad y son la base de lenguajes modernos como C++, Java o C#.

---

<sup>3</sup> Lenguaje de programación PHP <<http://es.wikipedia.org/wiki/PHP>>>

SmallTalk también permitió generar múltiples ventanas en una misma pantalla. El MVC es un patrón de Diseño que considera dividir una aplicación en tres módulos claramente identificadas y definidos funcionalmente: el modelo, las vistas y el controlador.

**El modelo:** Es un conjunto de clases que representan la información del mundo real que el sistema debe procesar, sin tener en cuenta la forma en que la información va a ser presentada ni los mecanismos que hacen que esos datos estén dentro del modelo, es decir sin tener relación con ninguna otra entidad dentro de la aplicación.

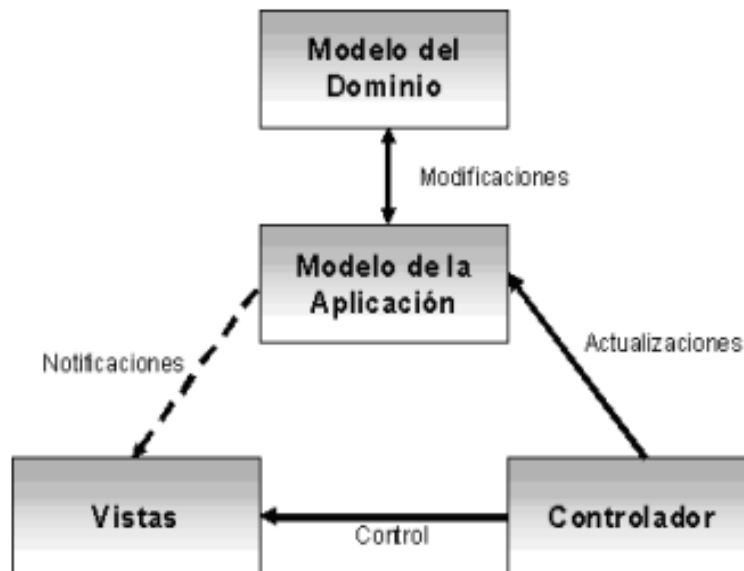
El modelo desconoce la existencia de las vistas y el controlador.

**Las vistas:** Son el conjunto de clases que se encargan de mostrar al usuario la información contenida en el **modelo**. Una vista está asociada a un modelo, pueden existir varias vistas asociadas al mismo modelo.

Una vista obtiene del modelo solamente la información que necesita para desplegar y se actualiza cada vez que el modelo cambia por medio de notificaciones.

**El controlador:** Es un objeto que se encarga de dirigir el flujo del control de la aplicación debido a mensajes externos, como datos introducidos por el usuario u opciones del menú seleccionadas por él. A partir de este mensaje el controlador se encarga de modificar el modelo o de abrir y cerrar las vistas.

El controlador tiene acceso al modelo y a las vistas pero el modelo y las vistas no tienen conocimiento de la existencia del controlador. (*Ver figura 2*).



**Figura 2: Relación entre módulos del patrón MVC<sup>4</sup>**

La construcción de aplicaciones Web utilizando PHP + MYSQL basadas en el patrón Modelo - Vista - Controlador. Este modelo separa las aplicaciones en 3 capas (por cuestiones de rendimiento, productividad del desarrollador o escalabilidad entre otras) delegando la responsabilidad de funciones de la siguiente manera:

**Capa modelo.** Implementa los objetos de negocio y es la responsable de la persistencia y la interacción con la base de datos. En el entorno que se va a aprender a utilizar este papel lo desarrolla el FRAMEWORK (un marco de trabajo realizado íntegramente en PHP OOP que permite desarrollar nuestros componentes de negocio).

**Capa vista.** Implementa la interface de usuario. En el entorno se va a desarrollar esta capa mediante paginas tpl, archivos JavaScript, y hojas de estilos, mediante librerías adaptadas al FRAMEWORK

**Capa control.** Es la responsable de controlar el flujo de navegación y de redireccionar las peticiones del usuario. Obviamente en una aplicación PHP OOP

<sup>4</sup>Bascon, E. . *El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing*. Acta Nova, 2004.



que siga el patrón MVC la comunicación entre capas es esencial. La capa control en este aspecto juega un papel fundamental puesto que es la encargada de recibir las peticiones de la interface de usuario y llamar al modelo para ejecutar reglas de negocio e interactuar con la base de datos. En el entorno se va a entrar en contacto con el controller. (*Ver figura 3*).

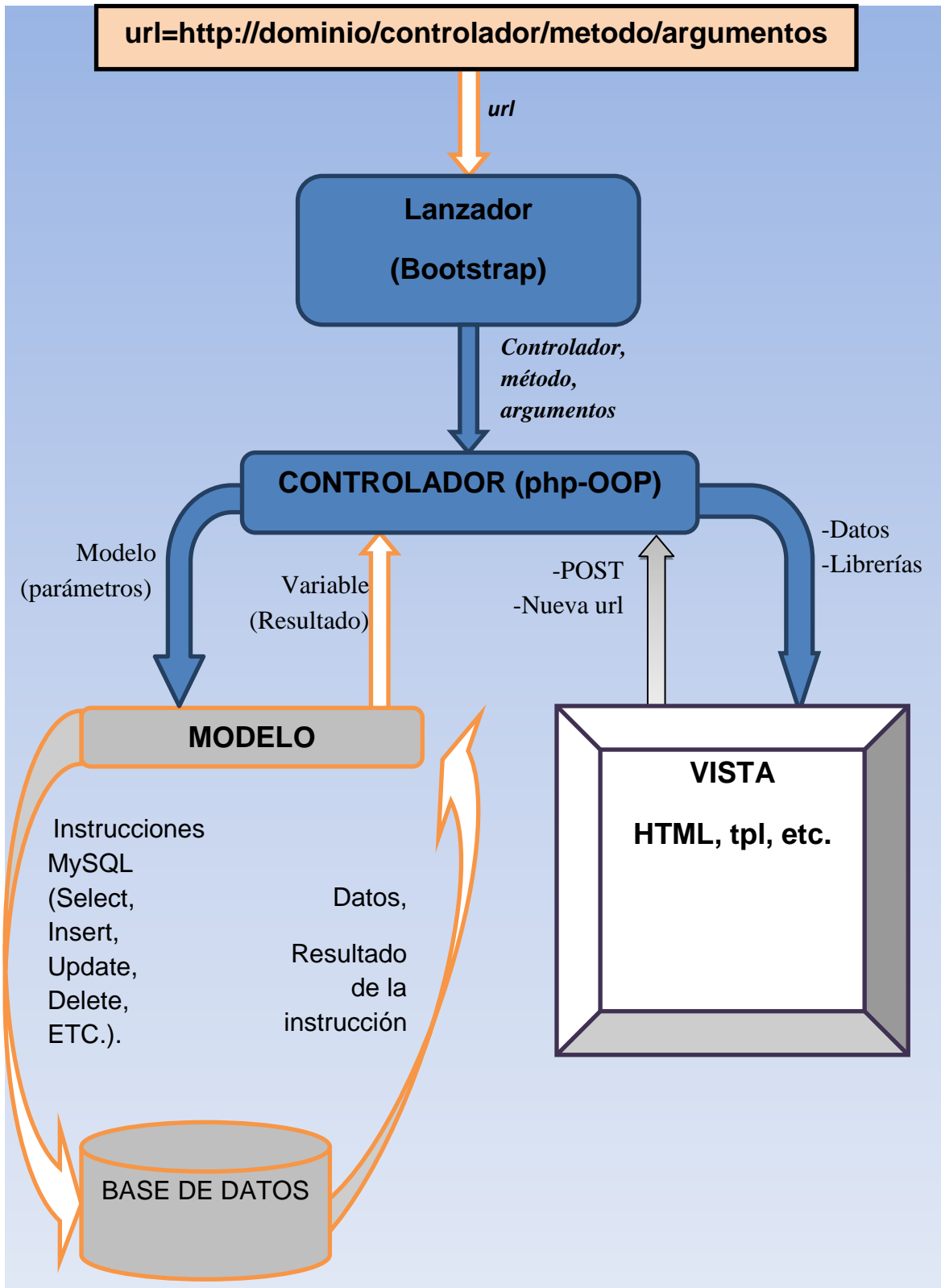


Figura 3: MODELO MVC SIGLHA- (MODELO-CONTROLADOR-VISTA)

## 1.4 FRAMEWORK SIGLHA

Este es un FRAMEWORK BASICO, creado y desarrollado por el autor de este proyecto, siguiendo los diferentes video tutoriales de <http://www.dlancedu.com/> , el cual corresponde a la creación de un FRAMEWORK BASICO.

Desarrollado en PHP 5 con programación orientada a objetos, con el motor de base de datos MYSQL 5 y funciona en un servidor web apache o IIS, usa el modelo MVC para la implementación de las clases, incluye librerías como JQUERY, SMARTY, JPGRAPH, PHPMAILER, FPDF; hojas de estilos CSS. Lo que lo convierte en una muy buena herramienta base para desarrollar aplicaciones web sin tener que recurrir a FRAMEWORS más robustos los cuales toca realizar un estudio previo de funcionamiento, es decir toca mirar el manual de uso y configuración tarea que resulta tediosa, mientras que el framework de SIGLHA es mucho más simple y sencillo de entender.

**Definición.** Es un FRAMEWORK PHP para desarrollar aplicaciones Web. El mismo permite la máxima reutilización en la programación Web y puede acelerar el proceso de desarrollo.

**Requerimientos.** Para correr una aplicación Web, usted necesita tener un servidor Web con soporte PHP 5.1.0 o superior. Para desarrolladores que deseen utilizar este framework, el entendimiento de Programación Orientada a Objetos (OOP) será de gran ayuda ya que este es un *framework* totalmente basado en OOP.

**¿Por qué utilizarlo?** Es un *framework* genérico de programación Web que puede ser utilizado para todo tipo de aplicaciones Web. Gracias a que es liviano de correr y está equipado con buenas soluciones, es adecuado para desarrollar aplicaciones web, Sistemas de comercio electrónico.

**Frente a otros *frameworks*?** Académicamente es muy importante al ser un framework básico da la posibilidad de añadirle librerías propias, para poder convertirlo en un framework universitario robusto que sirva de ayuda y base para desarrollar proyectos futuros.

Para crear aplicaciones utilizando este FRAMEWORK, debe seguir un simple proceso compuesto de 3 pasos:

1. Crear la base de datos.
2. Generar el código PHP de la base
3. Modificar el código para adecuarlo a sus necesidades.

Estructura del Framework Básico (ver figura 4)

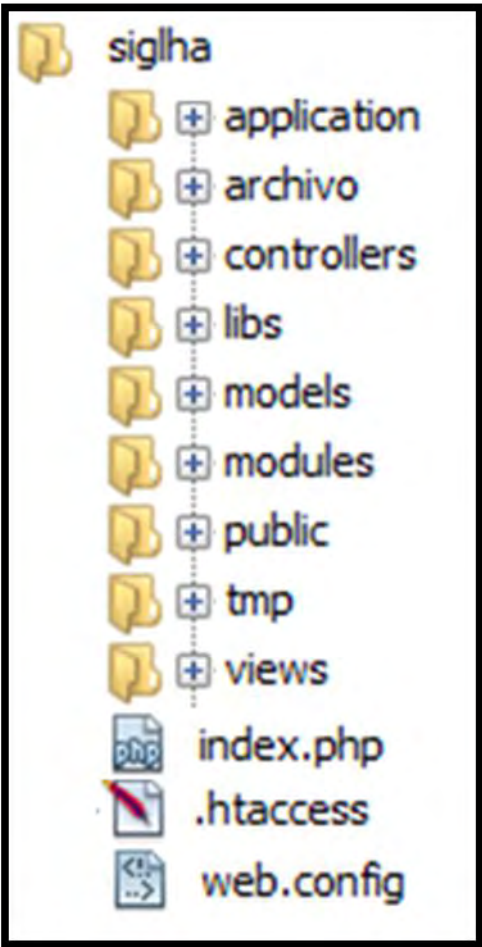
ESQUEMA	DESCRIPCION
 <p>El diagrama muestra la estructura de archivos del Framework Básico. El directorio principal es 'siglha'. Dentro de 'siglha' hay subdirectorios: 'application', 'archivo', 'controllers', 'libs', 'models', 'modules', 'public', 'tmp' y 'views'. Además, hay archivos: 'index.php', '.htaccess' y 'web.config'.</p>	<ul style="list-style-type: none"> <li>- <b>Application:</b> Carpeta que aloja las clases necesarias para el funcionamiento del Framework</li> <li>- <b>Archivo:</b> Carpeta para guardar archivos creados por el sistema: bitácora, backups, etc.</li> <li>- <b>Controllers:</b> Carpeta que contiene las diferentes clases controladoras del sistema</li> <li>- <b>Libs:</b> el framework usa librerías externas para hacer más eficiente la programación, dichas librerías pueden ser: JQuery, SMARTY, JPGRAPH, PHPMAILER, etc.</li> <li>- <b>Models:</b> Carpeta que aloja las clases modelo del sistema</li> <li>- <b>Modules:</b> si se requiere programar módulos complejos en esta carpeta se crean, ellas poseen sus propios controladores, módulos y vistas.</li> <li>- <b>Public:</b> carpeta para almacenar los archivos multimedia usados por la interfaces de usuarios, al igual de archivos que pueden ser descargados por los usuarios, ej.: archivos multimedia usados por la aplicación.</li> <li>- <b>Tmp:</b> carpeta necesaria por la librería SMARTY para crear archivos temporales.</li> <li>- <b>Views:</b> carpeta que contiene los archivos vista o archivos que son encargados de mostrar la interface a los usuarios</li> <li>- <b>Index:</b> Archivo principal cargado inicialmente al solicitar la página.</li> <li>- <b>.htaccses:</b> archivo para configurar la captura de la URL en una variable si el servidor usa Apache como servidor web</li> <li>- <b>Web.config:</b> archivo para configurar la captura de la URL en una variable si el servidor usa IIS como servidor web</li> </ul>

Figura 4: Sistema de archivos FRAMEWORK

## 1.5 PROGRAMACION EXTREMA<sup>5</sup>

**1.5.1 Introducción A Programación Extrema.** Esta es una de las metodologías de desarrollo de software más utilizada en la calidad, para proyectos de corto plazo y corto equipo. Consisten en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final como uno de los requisitos para llegar al éxito del proyecto.

El objetivo que se persigue con esta metodología es el desarrollo de software sea una actividad que no se concentre tanto que la documentación y que reduzca la complejidad del diseño de los sistemas. Esta es una metodología de desarrollo ágil basada en una serie de valores y prácticas o principios que pretenden incrementar la productividad los desarrolladores y programadores al realizar proyectos de software. Le exige que se le ocurrió hace

En esta metodología, el cliente se convierte en un miembro más de equipo de trabajo y es el encargado de decidir que se implementa, puede añadir, cambiar o quitar requerimientos en cualquier momento para lo cual debe estar enterado constantemente del estado real y el progreso del proyecto obteniendo lo máximo de cada semana de trabajo.

La programación extrema está guiada por una serie de valores que rigen el proceso de análisis, diseño e implementación del software. Estos valores son los siguientes: comunicación, simplicidad, retroalimentación, coraje, y respeto.

- **Comunicación:** comunicación entre programadores (o equipo de desarrollo) y entre estos y el cliente. Un aspecto de la comunicación sea realizar cuando los programadores se intercambian el código. Para que esto se pueda hacer, es necesario que éste sea sencillo y las clases estén bien definidas. Además, en lugar de utilizar comentarios en el código, se debe usar código auto documentado, pues éste no queda desfasado si se realiza alguna modificación posterior.

La comunicación entre cliente y el equipo de trabajo. Esta línea de comunicación con el cliente se realiza para la definición de los requerimientos del sistema que se desarrollará. En este proceso, el cliente determina cuál es la prioridad para el desarrollo de cada una de estas historias.

---

<sup>5</sup> BECK, KENT. *Una Explicación de la Programación Extrema: Aceptar el cambio*. Madrid: Addison-Wesley Iberoamericana Espanya, S.A. Todo el libro, 2002.

- **Simplicidad:** lo que busca la programación extrema es hacer un diseño simple del programa que se desee desarrollar, para que se hagan más sencillas las labores de mantenimiento. Una de las cosas que más complican el mantenimiento de una aplicación es el que el código fuente sea difícil de entender y modificar. Pues, cada vez que se lo modifique tiende a volverse más complicado de reparar.

Una de las herramientas adecuadas para evitar que los códigos complejos invadan la aplicación, consiste en realizar adecuadas documentaciones de lo que hacen cada una de las clases y métodos, Además, dar un buen nombre a las clases, métodos y variables contribuye a que el código sea entendible, sin tener miedo a que el poner nombres largos retrase el rendimiento del sistema.

- **Retroalimentación:** en un desarrollo de software en el que se utilice la programación extrema, debe existir una comunicación fluida entre el cliente y el equipo de trabajo, de tal manera que uno aporte datos de interés al otro y viceversa, es esto consiste la retroalimentación.

Al realizar estos intercambios frecuentes de información, fácilmente se pueden corregir los errores que se presenten, sin tener que esperar a que el proyecto esté tan avanzado y sea más complicado realizar modificaciones, cuando se realizan ciclos y que el trabajo de los desarrolladores no se desperdicie.

La retroalimentación también se realiza en el proceso de codificación. Al realizar pruebas unitarias de cada una de las clases que se programen, se pueden corregir los errores de implementación y ajustar algunas falencias en el diseño

- **Coraje:** la aplicación exacta de esta metodología exige algo de coraje, pues plantea algunas acciones que normalmente no son aceptadas en el mundo del desarrollo del software. Por ejemplo, uno de los principios de esta metodología es que se trabaje programando en parejas, Este principio obliga a utilizar dos empleados por equipo de cómputo. Muchas industrias no lo aceptan pues asumen que reduce la productividad laboral.

En este caso, el coraje que se desplieguen los desarrolladores contribuiría a que se libren de los temores infundados de que la programación en parejas reduce la productividad del equipo de desarrollo. Este temor se reduce cuando se considera que al programar de esta manera se reducen los errores fruto del cansancio o la desconcentración que pueda tener un solo programador. En efecto, la reducción de errores contribuye en buena medida al aumento de la productividad.

El coraje también se ve reflejado en la aplicación de otros valores como la simplicidad y la retroalimentación, en el caso de la simplicidad, al tomar la determinación de no codificar requerimientos no funcionales, antes de terminar el desarrollo del proyecto. Esto es importante, pues muchos programadores desean añadir a sus aplicaciones algunos detalles que la hagan atractiva, pero al “distraerse” con la programación de módulos grandes con la incertidumbre de que no estén a la altura de los requisitos planteados.

- **Respeto:** el respeto es fundamental en esta metodología de desarrollo, y se manifiesta de varias formas. Una manera consiste en respetar el trabajo realizado por los compañeros de desarrollo, de tal manera que no se hacen cambios innecesarios en el código fuente escrito por alguien del grupo o por la pareja de programación.

Al realizar estos cambios a última hora puede generar que las pruebas realizadas de alguna de las clases ya no conserve la aceptación que se le debió dar en una primera instancia. Por otro lado, con esta metodología de trabajo se genera un ambiente laboral que contribuye a la satisfacción de los integrantes del grupo, pues el trabajo de cada uno es valorado y respetado.

Estos valores deben guiar la aplicación de las etapas o fases de la metodología. Estas fases son: Planificación, diseño, Desarrollo y Pruebas. A su vez, cada fase está dividida en varias etapas, en los siguientes ítems se analiza cada una de las fases con sus derivados.

**1.5.2 Fase de planificación.** En esta fase se realiza el análisis de todos los requerimientos del proyecto. Se establece los requisitos de la aplicación en compañía del cliente. Se realizan reuniones periódicas con el fin de determinar el tiempo de desarrollo, el plan de entregas y las iteraciones que se realizan. Los requisitos que se establecen en esta fase son redactados por el cliente y se denominan historias de usuario. Cada una de las cuales se codifica en determinado lapso de tiempo y permite evaluar el progreso de la implementación.

Esta metodología posee las siguientes características o subprocesos: Redactar las historias de usuario, Realizar plan de entregas, Planificar las iteraciones, Medir la velocidad del proyecto, Programar en parejas y Hacer reuniones diarias.

Dado que, en esta metodología, no todo los subprocesos de las fases genera documentación para presentar en los resultados del proyecto, en los siguientes párrafos, de esta y de las demás fases, sólo se describen las etapas que si deben documentarse.

Historias de usuario. Las historias de usuarios tienen la misma finalidad que los casos de uso del proceso unificado, pero con algunas diferencias: son descripciones narradas por el cliente en unas cuantas ideas que puntualizan lo que se quiere desarrollar en una sola sección del sistema. Como es de esperarse, en estas descripciones no se aportan detalles de implementación, ni lenguajes, ni nombres de controles, ni aspectos técnicos, ni nada por el estilo en cuanto a ingeniería. Esto debido a que las narraciones no son provistas por el equipo de desarrollo

Las historias de usuarios permiten calcular, de manera general, cuál es el tiempo de desarrollo (pues cada historia debería ser implementada en un lapso de una a tres semanas), a la vez que guían el proceso de implementación. Por otro lado, contribuyen a evaluar el progreso de proyecto. Cada que se implementa una historia de usuario, se debe aplicar una prueba que determine si ésta se efectuó de forma exitosa, tal como el cliente la concibió inicialmente.

En cuanto al cronograma a seguir para la implementación de las historias de usuario, el equipo de desarrollo debe realizar varias reuniones periódicas con el cliente y determinar cuántas historias se codifican en cada periodo de tiempo.

Este plan de desarrollo recibe el nombre de “Plan de Entregas”, en el cual cada sección del plan de entregas se convierte en una iteración, de duración aproximada de semanas.

Con este plan de entregas y la determinación de las “iteraciones” se va evaluando el progreso del proyecto para ver si avanza a la velocidad esperada. Esta evaluación se realiza cada cierto tiempo en las “Reuniones diarias” que lleva a cabo el equipo de trabajo, en donde todos deben tener voz y voto.

**1.5.3 Fase de diseño.** Luego de redactar las historias de usuario, la metodología plantea la creación de Tarjetas CRC. Estas tarjetas se redactan en base al código fuente que tendría la aplicación. Por cada clase de código que se pretenda programar (con esto se deduce tras un análisis de diseño) se realiza una Tarjeta CRC. Luego, con estas tarjetas se evalúa que los programadores implementen el programa tal como fue analizado por el grupo de trabajo. Esta evaluación se realiza con la aplicación de “Pruebas de Unidad” a cada una de las tarjetas CRC.

Además de las tarjetas CRC, esta etapa de la metodología sugiere que la creación de una descripción general del funcionamiento básico del sistema, conocida como “Metáfora del Sistema”, en la que se detallan los aspectos técnicos del diseño de la aplicación. Este diseño contribuye a la creación de las tarjetas CRC.

Por otro lado, también se espera que los desarrolladores creen “Soluciones Puntuales”, en las que se realicen programas que cumplan con los requisitos



esperados y sean lo menos complicadas posibles. Esto se consigue con la “Funcionalidad Mínima” que consiste en concentrarse estrictamente en los requerimientos del usuario, en cada historia de usuario, y no dedicar tiempo a programar detalles que el usuario final quizá ni utilice.

Finalmente, en esta etapa del diseño, se espera que los desarrolladores realicen un “Reciclaje” continuo, pero eficaz, del código fuente, tanto de otros proyectos, como del que se está implementando. Esto permite que se puedan aprovechar mejor las funcionalidades que ya se hayan implementado. Se dice que este proceso debe ser eficaz en el sentido de que debe garantizarse que se cumplan las funcionalidades esperadas, pues un código de otro proyecto u otra sección del mismo, normalmente no encaja perfectamente en otro lugar, este proceso se denomina “Refactorizar”.

**Tarjetas CRC:** las iniciales “CRC” significan Clases, responsabilidades y colaboraciones. Como se mencionó, éstas representan objetos y las clases que los definen. Al diseñar utilizando estas tarjetas, es posible que los programadores eviten el antiguo método de programación basado en procedimientos específicos. En lugar de eso, se realiza un diseño orientado a objetos, que prevalece y el más fácil de mantener y utilizar.

En cada tarjeta CRC, se debe incluir la siguiente información: el nombre de la Clase, las Responsabilidades que le han asignado (esto es, los métodos que se implementan) y las Colaboraciones, es decir, las clases que se encargan de asistirle en la ejecución de algunas de sus responsabilidades.

En la siguiente ilustración se muestra un ejemplo de lo que sería una tarjeta CRC.  
(Ver figura 5)

Nombre del Proyecto: \_\_\_\_\_

Nombre de la Clase: \_\_\_\_\_

Responsabilidades (qué debe hacer)	R	Clases Asistentes (que colaboran)
<b>Desarrollador</b>		
<b>Observaciones:</b>		

**Figura 5. Ejemplo del formato de una tabla CRC<sup>6</sup>**

<sup>6</sup> BECK, KENT. *Una Explicación de la Programación Extrema: Aceptar el cambio*. Madrid: Addison-Wesley Iberoamericana Espanya, S.A. Todo el libro, 2002.

**1.5.4 Fase de desarrollo.** A diferencia de otras metodologías en las que, al iniciar la fase de desarrollo(o su equivalente), el cliente tiene poca participación y poca interacción con el equipo de trabajo, en XP se plantea todo lo contrario. La “Disponibilidad del cliente” debe ser continua y total, pues se lo considera un miembro más del equipo de desarrollo, por lo que contribuye a detallar las historias de usuario y a evaluar su implementación.

Otra faceta de esta fase de desarrollo, es la programación en parejas. Como se ha explicado en otras secciones, hacerlo de esta manera beneficia al grupo de trabajo pues dos personas programando sobre una misma clase evitan más errores que un solo programador, que por lo general comete muchos debido a la presión laboral.

Con el fin de evaluar la codificación de las Tarjetas CRC y la implementación de las historias de usuario, en esta fase de la metodología aparece una nueva herramienta, las “Unidades de prueba”. Estos test llegan a ser el elemento principal de esta fase de la metodología.

Unidad de pruebas: los test de evaluación prueban el funcionamiento de los códigos que se vayan a implementar según se haya planteado en las tarjetas CRC, en la fase de diseño. Crear estos test antes contribuye a saber qué es exactamente lo que tiene que hacer el código a implementar y así evitar divagar en la codificación y concentrarse en la realización de las funcionalidades específicas del proyecto.

En algunos casos, en los que las tarjetas CRC describan muchas funcionalidades, es posible fragmentar las responsabilidades que debe cumplir una tarea a programar en pequeñas unidades, de esta forma se crean primero los test para cada unidad y a continuación se desarrolla dicha unidad, así poco a poco consigue un desarrollo que cumpla todos los requisitos especificados.

**1.5.5 Fase de pruebas.** Las unidades o test de pruebas son parte indispensable de la programación Extrema. Estas se convierten en una herramienta de desarrollo y material de apoyo, mas no en un paso de verificación que puede despreciarse por más que parezca que el código esté fusionando correctamente.

Es por esto que la metodología XP exige constantes pruebas en el desarrollo, ya que descubrir todos los errores que puedan presentarse lleva tiempo y más se si dejan para el final del proyecto. En esta fase es importante tener en cuenta las siguientes características: implantación y pruebas de aceptación.

**Implementación:** el proceso de implantación se realiza luego de garantizar que el código se encuentra completo. Lo que quiere decir que se han realizado las unidades de pruebas y se ha hecho una revisión de los requerimientos planteados por el cliente, que aseguran que el código está de acuerdo con la solicitud. El

código debe ser implantado cuando supere sus correspondientes test de unidades.

Un punto importante es usar test de unidad (que llegan a ser test de implementación y no de aceptación) que no tengan ninguna dependencia del código que en un futuro evaluará, Hay que crear los test abstrayéndose del futuro código, de esta forma se asegura la independencia del test respecto al código que evalúa.

Los distintos test se deben subir al repositorio de código acompañados del código que verifican. Ningún código puede ser publicado en el repositorio sin que haya pasado su test de funcionamiento, de esta forma se asegura el uso colectivo del código (explicado en el apartado anterior).

**Pruebas de aceptación:** los test mencionados anteriormente sirven para evaluar las distintas tareas en las que ha sido dividida una historia de usuario. Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear “Pruebas de aceptación”.

Estas pruebas son creadas y usadas por los clientes para comprobar que las distintas historias de usuario cumplen su cometido.

Las pruebas de aceptación están basadas en las historias de usuario. Se hace un planteamiento de los puntos del desarrollo que deben probarse para corroborar que funcionen correctamente. Es para esto que se crean las pruebas de aceptación, determinando cuales son los aspectos que el cliente desea revisar y así garantizar su correcto funcionamiento. Por ello la importancia de crear cuantas pruebas de aceptación sean necesarias para que todo el trabajo realizado cuente con la calidad que exige.

Es fundamental desarrollar las pruebas de aceptación, porque es de esta forma que se demuestra el progreso y el proceso que se sigue en la ejecución del plan de iteración es para las historias de usuario, ya que estas no se consideran completas hasta que o superan sus pruebas de aceptación.

## 2. FASE DE PLANEACION

En esta fase se realizaron reuniones con los usuarios del sistema, para recolectar la información necesaria para el análisis y desarrollo del proyecto. Las historias a continuación son el resultado narrado de cómo funciona el sistema de apadrinamiento de parte de los usuarios encargados de las diferentes actividades, de estas historias nacen los tiempos de programación, la identificación de los procesos, el flujo de información, las relaciones entre procesos y usuarios, etc.

### 2.1 HISTORIAS DE USUARIOS Nº1: FUNCIONAMIENTO EN LA WEB

**Tabla 1: Historias de Usuarios Nº1: Funcionamiento en la Web**

Historia de Usuario Nº1	
Título de la Historia	Funcionamiento en la WEB
Módulo de la Historia	General
Código de la Historia	HU – GR- 001
Descripción	
Los usuarios pueden tener acceso a SIGLHA desde cualquier dispositivo con internet y que posea un browser o navegador.	
Observaciones	
Estado de implementación	Implementada

### 2.2 HISTORIAS DE USUARIOS Nº2: USO DEL SISTEMA

**Tabla 2: Historias de Usuarios Nº2: Uso del Sistema**

Historia de Usuario Nº2	
Título de la Historia	Uso del sistema
Módulo de la Historia	General
Código de la Historia	HU – GR- 002
Descripción	
Se debe contar con un nombre de usuario y contraseña, acceso a internet y se	

debe usar en lo preferible GOOGLE CHROME pues en la programación se hizo uso del plugin html5 que este posee para la validación local de los datos antes de ser enviados al servidor	
Observaciones	
Estado de implementación	Implementada

### 2.3 HISTORIAS DE USUARIOS N°3: BASE DE DATOS

**Tabla 3: Historias de Usuarios N°3: Base de Datos**

Historia de Usuario N°3	
Título de la Historia	Base de Datos
Módulo de la Historia	General
Código de la Historia	HU – GR- 003
Descripción	
La base de datos usada es MYSQL 5.5	
Observaciones	
Estado de implementación	Implementada

### 2.4 HISTORIAS DE USUARIOS N° 4: LISTAS DE CONTROL DE ACCESO – ACL

**Tabla 4: Historias de Usuarios N° 4: Listas de Control de Acceso – ACL**

Historia de Usuario N°4	
Título de la Historia	Listas de Control de Acceso – ACL
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU – AU- 004
Descripción	
<p>Método implementado para asegurar que no todos los usuarios tengan acceso a todos los módulos y no solo a los funcionarios si no a personas y ataques al sistema.</p> <p>Implementando un sistema de permisos y roles a cada método o función de SIGLHA se le asigna un permiso de esta manera cada acceso es evaluado primero por ACL</p>	

Observaciones	
Estado de implementación	Implementada

## 2.5 HISTORIAS DE USUARIOS Nº 5: CREAR DE ROLES

**Tabla 5: Historias de Usuarios Nº 5: Crear de Roles**

Historia de Usuario Nº5	
Título de la Historia	Crear Roles
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 005
Descripción	
Crea un nuevo role que hará parte del sistema	
Observaciones	
Estado de implementación	Implementada

## 2.6 HISTORIAS DE USUARIOS Nº 6: EDITAR ROLE

**Tabla 6: Historias de Usuarios Nº 6: Editar Role**

Historia de Usuario Nº6	
Título de la Historia	Editar Role
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 006
Descripción	
Cambia el nombre de un role	
Observaciones	
Estado de implementación	Implementada

## 2.7 HISTORIAS DE USUARIOS Nº 7: LISTAR ROLE

**Tabla 7: Historias de Usuarios Nº 7: Listar Role**

Historia de Usuario Nº7	
Título de la Historia	Listar Role
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 007
Descripción	
Lista todos los roles existentes y da opciones para agregarles o modificarles permisos	
Observaciones	
Estado de implementación	Implementada

## 2.8 HISTORIAS DE USUARIOS Nº 8: CREAR PERMISOS

**Tabla 8: Historias de Usuarios Nº 8: Crear Permisos**

Historia de Usuario Nº8	
Título de la Historia	Crear Permisos
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 008
Descripción	
Este método crea un permiso que debe ir relacionado a los métodos programados, y ligados directamente al código fuente.	
Observaciones	
Estado de implementación	Implementada

## 2.9 HISTORIAS DE USUARIOS Nº 9: LISTAR PERMISOS

**Tabla 9: Historias de Usuarios Nº 9: Listar Permisos**

Historia de Usuario Nº9	
Título de la Historia	Listar Permisos
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 009



Descripción	
Muestra en una lista todos los permisos que pueden ser asignados a los roles creados	
Observaciones	
Estado de implementación	Implementada

## 2.10 HISTORIAS DE USUARIOS Nº 10: ASIGNAR PERMISOS A ROLE

**Tabla 10: Historias de Usuarios Nº 10: Asignar permisos a Role**

Historia de Usuario Nº10	
Título de la Historia	Asignar permisos a Role
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 010
Descripción	
Cuando se crea un role este no tiene permisos asignados, así que este método asigna a cada role los permisos que el administrador le asigne.	
Observaciones	
Estado de implementación	Implementada

## 2.11 HISTORIAS DE USUARIOS Nº 11: CREAR USUARIO

**Tabla 11: Historias de Usuarios Nº 11: Crear Usuario**

Historia de Usuario Nº11	
Título de la Historia	Crear Usuario
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 011
Descripción	
Crea o agrega un usuario al sistema, asignándole nombre de usuario y contraseña y un role para acceder a las funcionalidades del sistema	
Observaciones	
Estado de implementación	Implementada

## 2.12 HISTORIAS DE USUARIOS Nº 12: EDITAR DATOS USUARIO

**Tabla 12: Historias de Usuarios Nº 12: Editar Datos Usuario**

Historia de Usuario Nº12	
Título de la Historia	Editar Datos Usuario
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 012
Descripción	
La información de los usuarios puede cambiar no solo la personal sino también las contraseñas, este método permite corregir y cambiar los datos	
Observaciones	
Estado de implementación	Implementada

## 2.13 HISTORIAS DE USUARIOS Nº 13: EDITAR PERMISOS USUARIO

**Tabla 13: Historias de Usuarios Nº 13: Editar Permisos Usuario**

Historia de Usuario Nº13	
Título de la Historia	Editar Permisos Usuario
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 013
Descripción	
Cambia los permisos asignados por defecto al momento de asignar un role, permitiendo restringir el acceso o por el contrario dando privilegios a módulos no permitidos.	
Observaciones	
Estado de implementación	Implementada

## 2.14 HISTORIAS DE USUARIOS Nº 14: LISTAR USUARIO

**Tabla 14: Historias de Usuarios Nº 14: Listar Usuario**

Descripción	
Historia de Usuario Nº14	
Título de la Historia	Listar Usuario
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 014
Descripción	
Muestra los usuarios que hacen parte del sistema	
Observaciones	
Estado de implementación	Implementada

## 2.15 HISTORIAS DE USUARIOS Nº 15: ELIMINAR USUARIO

**Tabla 15: Historias de Usuarios Nº 15: Eliminar Usuario**

Historia de Usuario Nº15	
Título de la Historia	Eliminar Usuario
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 015
Descripción	
Esta opción no está permitida si el usuario ya realizó actividades en SIGLHA pues deja referencia en las tablas a las cuales afecto, y para poder eliminarlo se debería eliminar las referencias, por eso se creó la opción de poner en inactivo al usuario	
Observaciones	
Estado de implementación	Implementada

## 2.16 HISTORIAS DE USUARIOS Nº 16: CREAR COPIAS DE SEGURIDAD

**Tabla 16: Historias de Usuarios Nº 16: Crear copias de seguridad**

Historia de Usuario Nº16	
Título de la Historia	Crear copias de seguridad
Módulo de la Historia	Administración/Usuarios
Código de la Historia	HU- AU- 016
Descripción	
Es importante realizar copias de seguridad a la base de datos para evitar pérdida de la información así que se implementa este método para descargar una copia ya sea de la estructura, de una tabla en especial o de toda la base de datos para poder en algún momento y si ocurre algo extraordinario que elimine la base poder restaurarla con este archivo.	
Observaciones	
Genera un archivo .sql y para restaurarlo se hace directamente del gestor de la base de datos	
Estado de implementación	Implementada

## 2.17 HISTORIAS DE USUARIOS Nº 17: CREAR UBICACIONES

**Tabla 17: Historias de Usuarios Nº 17: Crear Ubicaciones**

Historia de Usuario Nº17	
Título de la Historia	Crear Ubicaciones
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 017
Descripción	
El sistema debe permitir agregar barrios o poblaciones las cuales servirán como ubicación para los niños, familiares, etc.	
Observaciones	
Estado de implementación	Implementada

## 2.18 HISTORIAS DE USUARIOS Nº 18: EDITAR UBICACIONES

**Tabla 18: Historias de Usuarios Nº 18: Editar Ubicaciones**

<b>Historia de Usuario Nº18</b>	
Título de la Historia	Editar Ubicaciones
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 018
Descripción	
Al ingresar una ubicación mal, o al cambiar el nombre el sistema debe permitir cambiar el nombre de la ubicación	
Observaciones	
Estado de implementación	Implementada

## 2.19 HISTORIAS DE USUARIOS Nº 19: ELIMINAR UBICACIONES

**Tabla 19: Historias de Usuarios Nº 19: Eliminar Ubicaciones**

<b>Historia de Usuario Nº19</b>	
Título de la Historia	Eliminar Ubicaciones
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 019
Descripción	
Si se presenta error al momento de ingresar una ubicación con este método se puede corregir	
Observaciones	
Estado de implementación	Implementada

## 2.20 HISTORIAS DE USUARIOS Nº 20: LISTAR UBICACIONES

**Tabla 20: Historias de Usuarios Nº 20: Listar Ubicaciones**

<b>Historia de Usuario Nº20</b>	
Título de la Historia	Listar Ubicaciones
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 020
Descripción	
Este método agrega las posibles ubicaciones que los niños, escuelas, etc. A las que pueden pertenecer para su fácil ubicación, para hacer estándar el uso de estos nombres y no tener datos ambiguos.	
Observaciones	
Estado de implementación	Implementada

## 2.21 HISTORIAS DE USUARIOS Nº 21: CONSULTAR UBICACIONES

**Tabla 21: Historias de Usuarios Nº 21: Consultar Ubicaciones**

<b>Historia de Usuario Nº21</b>	
Título de la Historia	Consultar Ubicaciones
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 021
Descripción	
Permite consultar las diferentes ubicaciones de una región en especial	
Observaciones	
Estado de implementación	Implementada

## 2.22 HISTORIAS DE USUARIOS N° 22: AGREGAR PADRINO

**Tabla 22: Historias de Usuarios N° 22: Agregar Padrino**

Historia de Usuario N°22	
Título de la Historia	Agregar Padrino
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 022
Descripción	
Ingresar nuevos padrinos a la base de datos	
Observaciones	
Estado de implementación	Implementada

## 2.23 HISTORIAS DE USUARIOS N° 23: EDITAR PADRINO

**Tabla 23: Historias de Usuarios N° 23: Editar Padrino**

Historia de Usuario N°23	
Título de la Historia	Editar Padrino
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 023
Descripción	
Este método será usado para cambiar datos del padrino ya sea en nombres o datos diferentes	
Observaciones	
Estado de implementación	Implementada

## 2.24 HISTORIAS DE USUARIOS N° 24: ELIMINAR PADRINO

**Tabla 24: Historias de Usuarios N° 24: Eliminar Padrino**

Historia de Usuario N°24	
Título de la Historia	Eliminar Padrino
Módulo de la Historia	Manejo y Control

Código de la Historia	HU- MC- 024
Descripción	
Tampoco está permitido borrar padrinos pero se deja la opción por si se ingresa un padrino erradamente.	
Observaciones	
Si el padrino tiene referencias en otras tablas no se podrá eliminar hasta que la referencia sea eliminada primero	
Estado de implementación	Implementada

## 2.25 HISTORIAS DE USUARIOS Nº 25: LISTAR PADRINOS

**Tabla 25: Historias de Usuarios Nº 25: Listar Padrinos**

Historia de Usuario Nº1	
Título de la Historia	Listar Padrinos
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 025
Descripción	
Lista los padrinos que fueron modificados en orden descendente, y lista los datos básicos de ellos	
Observaciones	
Estado de implementación	Implementada

## 2.26 HISTORIAS DE USUARIOS Nº 26: CONSULTAR PADRINOS

**Tabla 26: Historias de Usuarios Nº 26: Consultar Padrinos**

Historia de Usuario Nº26	
Título de la Historia	Consultar Padrinos
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 026
Descripción	
Consultar el estado y la existencia de un padrino en particular es importante para evitar duplicidades y uso de padrinos que ya no estén vinculados.	
Observaciones	
Estado de implementación	Implementada



## 2.27 HISTORIAS DE USUARIOS N° 27: AGREGAR INFORMACIÓN NIÑO

**Tabla 27: Historias de Usuarios N° 27: Agregar Información Niño**

Historia de Usuario N°27	
Título de la Historia	Agregar Información Niño
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 027
Descripción	
Este es para ingresar información faltante o nueva de los niños	
Observaciones	
Estado de implementación	Implementada

## 2.28 HISTORIAS DE USUARIOS N° 28: EDITAR INFORMACIÓN NIÑO

**Tabla 28: Historias de Usuarios N° 28: Editar Información Niño**

Historia de Usuario N°28	
Título de la Historia	Editar Información Niño
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 028
Descripción	
Esta es la opción para actualizar la información básica del niño pues con el tiempo va cambiando, básicamente es sobre este método que los técnicos trabajaran más que en cualquier otro método	
Observaciones	
Estado de implementación	Implementada

## 2.29 HISTORIAS DE USUARIOS N° 29: ELIMINAR INFORMACIÓN NIÑO

**Tabla 29: Historias de Usuarios N° 29: Eliminar Información Niño**

Historia de Usuario N°29	
Título de la Historia	Eliminar Información Niño
Módulo de la Historia	Manejo y Control

Código de la Historia	HU- MC- 029
Descripción	
Esta acción será solo realizada por el administrador del sistema, pues no se puede eliminar información sin autorización	
Observaciones	
Estado de implementación	Implementada

### 2.30 HISTORIAS DE USUARIOS Nº 30: LISTAR INFORMACIÓN NIÑO

**Tabla 30: Historias de Usuarios Nº 30: Listar Información Niño**

Historia de Usuario Nº30	
Título de la Historia	Listar Información Niños
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 030
Descripción	
Esta opción es para listar los últimos niños a los que se le modifico información	
Observaciones	
Estado de implementación	Implementada

### 2.31 HISTORIAS DE USUARIOS Nº 31: CONSULTAR INFORMACIÓN NIÑOS

**Tabla 31: Historias de Usuarios Nº 31: Consultar Información Niños**

Historia de Usuario Nº31	
Título de la Historia	Consultar Información Niños
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 031
Descripción	
Es importante no solo para apadrinamiento poder conocer el estado del niño si no, para los otros departamentos de global que a veces necesitan conocer si un niño en particular pertenece y si tiene o no padrino	

Observaciones	
Estado de implementación	Implementada

### 2.32 HISTORIAS DE USUARIOS Nº 32: CREAR APADRINAMIENTO

**Tabla 32: Historias de Usuarios Nº 32: Crear Apadrinamiento**

Historia de Usuario Nº1	
Título de la Historia	Crear Apadrinamiento
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 032
Descripción	
Asignar padrino a niños sin padrinos	
Observaciones	
Estado de implementación	Implementada

### 2.33 HISTORIAS DE USUARIOS Nº 33: EDITAR APADRINAMIENTO

**Tabla 33: Historias de Usuarios Nº 33: Editar Apadrinamiento**

Historia de Usuario Nº33	
Título de la Historia	Editar Apadrinamiento
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 033
Descripción	
Cambiar de padrino a un niño o de ahijado a un padrino, es importante para mantener las relaciones tanto de ahijado y padrino actualizado.	
Observaciones	
Estado de implementación	Implementada

## 2.34 HISTORIAS DE USUARIOS N° 34: ELIMINAR APADRINAMIENTO

**Tabla 34: Historias de Usuarios N° 34: Eliminar Apadrinamiento**

Historia de Usuario N°34	
Título de la Historia	Eliminar Apadrinamiento
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 034
Descripción	
Esta opción es poco probable que se realice pero más sin embargo se realizara solo por el administrador del sistema.	
Observaciones	
Se debe tener en cuenta que los datos están relacionados y no el sistema no permite eliminar un dato si este tiene referencias en otras tablas	
Estado de implementación	Implementada

## 2.35 HISTORIAS DE USUARIOS N° 35: LISTAR APADRINAMIENTOS

**Tabla 35: Historias de Usuarios N° 35: Listar Apadrinamientos**

Historia de Usuario N°1	
Título de la Historia	Listar Apadrinamientos
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 035
Descripción	
Es una manera de consulta pero va más encaminada a mostrar cuales fueron los últimos registros ingresados al sistema	
Observaciones	
Estado de implementación	Implementada

## 2.36 HISTORIAS DE USUARIOS N° 36: CONSULTAR APADRINAMIENTO

**Tabla 36: Historias de Usuarios N° 36: Consultar Apadrinamiento**

Historia de Usuario N°36	
Título de la Historia	Consultar Apadrinamiento
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 036
Descripción	
Consulta el estado de apadrinamiento de un niño o un padrino, para verificar que exista el apadrinamiento y si no, poder identificar niños disponibles al igual que padrinos	
Observaciones	
Estado de implementación	Implementada

## 2.37 HISTORIAS DE USUARIOS N° 37: AGREGAR DIBUJO

**Tabla 37: Historias de Usuarios N° 37: Agregar Dibujo**

Historia de Usuario N°37	
Título de la Historia	Agregar Dibujo
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 037
Descripción	
Este método llena los registros con los datos de los niños que realizaron dibujo y lo entregaron a los técnico	
Observaciones	
Estado de implementación	Implementada

## 2.38 HISTORIAS DE USUARIOS N° 38: EDITAR DIBUJO

**Tabla 38: Historias de Usuarios N° 38: Editar Dibujo**

Historia de Usuario N°38	
Título de la Historia	Editar Dibujo
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 038
Descripción	
Una manera de corregir información mal digitada es poder editar lo ingresado.	
Observaciones	
Estado de implementación	Implementada

## 2.39 HISTORIAS DE USUARIOS N° 39: ELIMINAR DIBUJO

**Tabla 39: Historias de Usuarios N° 39: Eliminar Dibujo**

Historia de Usuario N°1	
Título de la Historia	Eliminar Dibujo
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 039
Descripción	
Contemplando la idea que es un humano el que digita la información puede ocurrir que se equivoque en algún dato y pues el sistema debe permitir corregir dicho error con la eliminación de ese registro, claro está que solo lo podrá realizar un usuario con autorización	
Observaciones	
Estado de implementación	Implementada

## 2.40 HISTORIAS DE USUARIOS N° 40: LISTAR DIBUJO

**Tabla 40: Historias de Usuarios N° 40: Listar Dibujo**

Historia de Usuario N°1	
Título de la Historia	Listar Dibujo
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 040
Descripción	
Por medio de este método podremos ver los dibujos ingresados hasta el momento de la consulta	
Observaciones	
Estado de implementación	Implementada

## 2.41 HISTORIAS DE USUARIOS N° 41: CONSULTAR DIBUJO

**Tabla 41: Historias de Usuarios N° 41: Consultar Dibujo**

Historia de Usuario N°	
Título de la Historia	Consultar Dibujo
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 041
Descripción	
La campaña de dibujo es una de las campañas de fidelización más usadas en global así que es importante saber que niños faltan por realizar el dibujo.	
Observaciones	
Estado de implementación	Implementada

## 2.42 HISTORIAS DE USUARIOS N° 42: CONSULTAR ESTADO NIÑO

**Tabla 42: Historias de Usuarios N° 42: Consultar Estado Niño**

Historia de Usuario N°42	
Título de la Historia	Consultar Estado Niño
Módulo de la Historia	Manejo y Control
Código de la Historia	HU- MC- 042
Descripción	

El proceso de actualización es realizado según la antigüedad de los datos, en este orden de ideas es importante que el sistema verifique la antigüedad de la información y pueda recomendar que datos deben ser actualizados y cuáles no.	
Observaciones	
Estado de implementación	Implementada

## 2.43 HISTORIAS DE USUARIOS N° 43: EXPORTAR DATOS A ESPAÑA

**Tabla 43: Historias de Usuarios N° 43: Exportar Datos a España**

Historia de Usuario N°43	
Título de la Historia	Exportar Datos a España
Módulo de la Historia	Exportar
Código de la Historia	HU- EX -043
Descripción	
España pide una copia en Excel del estado de la base de datos de Colombia, así que es importante que el sistema pueda exportar los datos en Excel.	
Observaciones	
Estado de implementación	Implementada

## 2.44 HISTORIAS DE USUARIOS N° 44: REPORTAR BITÁCORA DE USO

**Tabla 44: Historias de Usuarios N° 44: Reportar Bitácora de Uso**

Historia de Usuario N°44	
Título de la Historia	Reportar Bitácora de Uso de SIGLHA
Módulo de la Historia	Reportes
Código de la Historia	HU- RI- 044
Descripción	
Para llevar un control y poder evaluar el uso del sistema de información es importante llevar un registro de actividades o transacciones realizadas por los usuarios del sistema, no solo para llevar un control, si no, que al presentarse perdidas de información por mala manipulación de los datos, poder identificar al responsable.	
Observaciones	
Estado de implementación	Implementada



### 3. FASE DE DISEÑO

En esta fase después de haber obtenido la información con las historias de usuario, están redactadas según el código fuente, aquí se definen bien las relaciones, colaboraciones y funciones de cada uno de los procesos a realizar en el sistema de información

#### 3.1 TARJETA CRC NO 1: NUEVO ROLE

**Tabla 45: Tarjeta CRC No 1: Nuevo Role**

TARJETA CRC No. 1	
Nombre de la Clase	ACLController
Nombre de la Función	Nuevo Role
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 001
Resumen	
Agregar un Nuevo Rol al sistema.	
Responsabilidades	
<ul style="list-style-type: none"><li>➤ Solicitar datos del role y permiso para la creación del nuevo role.</li><li>➤ Evaluar la inexistencia del Role.</li><li>➤ Asignar Role.</li><li>➤ Validar la información para el registro del Role.</li><li>➤ Informar el resultado de la operación</li></ul>	
Colaboraciones	
ACL Model: public function getRole Views: Nuevo_Role	
Estado de implementación	Implementada

#### 3.2 TARJETA CRC NO 2: EDITAR ROLE

**Tabla 46: Tarjeta CRC No 2: Editar Role**

TARJETA CRC No. 2	
Nombre de la Clase	ACLController
Nombre de la Función	Editar Role

Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 002
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Identificar Role a editar.</li> <li>➤ Modificar Role</li> <li>➤ Guardar cambios</li> <li>➤ Generar reporte de modificación.</li> <li>➤ Evaluar estado del usuario.</li> <li>➤ Editar permiso, datos de la cuenta y contraseña.</li> <li>➤ Validar información modificada.</li> <li>➤ Informar el resultado de la operación.</li> </ul>	
Colaboraciones	
ACL Model: Public Function EditarRole	
Views: Editar_role	
Estado de implementación	Implementada

### 3.3 TARJETA CRC NO 3: LISTAR ROLE

**Tabla 47: Tarjeta CRC No 3: Listar Role**

<b>TARJETA CRC No. 3</b>	
Nombre de la Clase	ACLController
Nombre de la Función	Index Role
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 003
Resumen	
Mostrar Lista de roles establecidos	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ identificar proceso a listar, Roles o permisos.</li> <li>➤ Visualizar Roles creados y sus respectivos permisos.</li> </ul>	
Colaboraciones	
ACL Model: Public Function Get Role	
Views: Index	
Estado de implementación	Implementada

### 3.4 TARJETA CRC NO 4: CREAR PERMISOS

**Tabla 48: Tarjeta CRC No 4: Crear Permisos**

TARJETA CRC No. 4	
Nombre de la Clase	ACLController
Nombre de la Función	Nuevo_ permiso
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 004
Resumen	
Generar un nuevo permiso para los usuarios.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información del permiso a crear.</li> <li>➤ Evaluar la inexistencia del permiso.</li> <li>➤ Generar permiso, clave del mismo y guardar cambios.</li> <li>➤ Validar la información.</li> <li>➤ Mostrar resultado de la operación.</li> </ul>	
Colaboraciones	
ACL Model: public function getPermisoNombre	
Views: Permisos	
Estado de implementación	Implementada

### 3.5 TARJETA CRC NO 5: LISTAR PERMISOS

**Tabla 49: Tarjeta CRC No 5: Listar Permisos**

TARJETA CRC No. 5	
Nombre de la Clase	ACLController
Nombre de la Función	Listar Permisos
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 005
Resumen	
Administración de permisos creados.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Mostrar todos los permisos ingresados al sistema.</li> </ul>	
Colaboraciones	
ACL Model: Public Function EditarRole	
Views: Editar_role	
Estado de implementación	Implementada

### 3.6 TARJETA CRC NO 6: ASIGNAR PERMISOS A ROLE

**Tabla 50: Tarjeta CRC No 6: Asignar permisos a Role**

TARJETA CRC No. 6	
Nombre de la Clase	ACL Controller
Nombre de la Función	permisos Role
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 006
Resumen	
Generar nuevo permiso a role.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información de Role y permiso a asignar.</li> <li>➤ Listar roles, identificar permisos otorgados y faltantes.</li> <li>➤ Asignar el nuevo permiso.</li> <li>➤ Guardar cambios realizados.</li> <li>➤ Generar reporte del resultado.</li> </ul>	
Colaboraciones	
ACL Model: public function getPermisosRole	
Views: Permiso_role	
Estado de implementación	Implementada

### 3.7 TARJETA CRC NO 7: CREAR USUARIO

**Tabla 51: Tarjeta CRC No 7: Crear Usuario**

TARJETA CRC No. 7	
Nombre de la Clase	RegistroController
Nombre de la Función	Index
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 007
Resumen	
Agregar nuevo usuario al sistema.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos personales al usuario y permiso para la creación del nuevo usuario</li> <li>➤ Asignar nombre de usuario, correo institucional, role y contraseña al usuario.</li> <li>➤ Validar la información para el registro del usuario.</li> <li>➤ Ingresar a la base de datos el nuevo usuario</li> <li>➤ Informar el resultado de la operación</li> </ul>	
Colaboraciones	
Registro Model: RegistrarUsuario	
Views: Index	
Estado de implementación	Implementada

### 3.8 TARJETA CRC NO 8: EDITAR DATOS DE USUARIO

**Tabla 52: Tarjeta CRC No 8: Editar Datos de Usuario**

TARJETA CRC No. 8	
Nombre de la Clase	RegistroController
Nombre de la Función	Editarusuario
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 008
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Listar usuarios creados.</li> <li>➤ Identificar role y estado del usuario.</li> <li>➤ Editar permiso, datos de la cuenta y contraseña.</li> <li>➤ Validar información modificada.</li> <li>➤ Informar el resultado de la operación.</li> </ul>	
Colaboraciones	
Registro Model: Public Function EditarUsuario	
Views: EditarUsuario	
Estado de implementación	Implementada

### 3.9 TARJETA CRC NO 9: EDITAR PERMISOS USUARIO

**Tabla 53: Tarjeta CRC No 9: Editar Permisos Usuario**

TARJETA CRC No. 9	
Nombre de la Clase	IndexController
Nombre de la Función	Permisos
Módulo del Sistema	Administración Usuarios /
Código de la Tarjeta CRC	CRC- AU- 009
Resumen	
Modificar permisos otorgados al usuario.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Identificar usuario y estado del usuario.</li> <li>➤ Listar permisos habilitados.</li> <li>➤ Modificar permisos.</li> <li>➤ Guardar cambios.</li> <li>➤ Generar reporte de resultados.</li> </ul>	
Colaboraciones	
IndexModel: Public Function GetPermisoUsuario	

Views: Permisos	
Estado de implementación	Implementada

### 3.10 TARJETA CRC NO 10: LISTAR USUARIO

**Tabla 54: Tarjeta CRC No 10: Listar Usuario**

TARJETA CRC No. 10	
Nombre de la Clase	RegistroController
Nombre de la Función	Index
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 010
Resumen	
Permite visualizar los usuarios creados y acceder a la opción editar los mismos.	
Responsabilidades	
➤ Mostrar todos los permisos ingresados al sistema.	
Colaboraciones	
RegistroModel: GetUsuarios Views: Index	
Estado de implementación	Implementada

### 3.11 TARJETA CRC NO 11: ELIMINAR USUARIO

**Tabla 55: Tarjeta CRC No 11: Eliminar Usuario**

TARJETA CRC No. 11	
Nombre de la Clase	RegistroController
Nombre de la Función	Eliminar
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 011
Resumen	
Eliminar Usuario creado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Listar Usuarios creados en el sistema.</li> <li>➤ Identificar usuario y seleccionar la opción eliminar.</li> <li>➤ Validar información.</li> <li>➤ Generar reporte de resultado.</li> </ul>	
Colaboraciones	
<b>Registro</b> Model: Public Function EliminarUsuario Views: Eliminar	
Estado de implementación	Implementada

### 3.12 TARJETA CRC NO 12: CREAR COPIAS DE SEGURIDAD

**Tabla 56: Tarjeta CRC No 12: Crear Copias de Seguridad**

TARJETA CRC No. 12	
Nombre de la Clase	BackupController
Nombre de la Función	Crear
Módulo del Sistema	Administración/Usuarios
Código de la Tarjeta CRC	CRC- AU- 012
Resumen	
Crear un Backup de todas las opciones de los diferentes módulos del sistema con estructura y/o Datos.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Seleccionar módulo de Administración/Usuario, Copias de Seguridad.</li> <li>➤ Listar elementos susceptibles de realizarles Backup.</li> <li>➤ Seleccionar la opción y la estructura de la tabla</li> <li>➤ Crear el backup.</li> <li>➤ Generar Backup.</li> </ul>	
Colaboraciones	
BackupModel: Public Function GetNameTablas	
Views: Crear	
Estado de implementación	Implementada

### 3.13 TARJETA CRC NO 13: CREAR UBICACIONES

**Tabla 57: Tarjeta CRC No 13: Crear Ubicaciones**

TARJETA CRC No. 13	
Nombre de la Clase	Ubicación Controller
Nombre de la Función	Nuevo o Insertar Ubicación
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 013
Resumen	
<p>En el módulo de manejo y control se encuentra la opción de lugares y en ella la de ubicación la cual permite listar las opciones ya ingresadas, así como crear nuevas opciones.</p> <p>La función insertar ubicación permite ingresar nuevas ubicaciones en el sistema de información según necesidad de la entidad.</p>	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información de la ubicación a crear.</li> </ul>	

<ul style="list-style-type: none"> <li>➤ Evaluar la inexistencia de la misma.</li> <li>➤ En agregar nueva ubicación listar las opciones permitidas, especificando en cada una la información solicitada.</li> <li>➤ Insertar la ubicación.</li> <li>➤ Generar reporte de resultado.</li> </ul>	
Colaboraciones	
<b>Ubicación</b> Model: Public Function <b>Insertar</b> Ubicacion	
Views: Nuevo	
Estado de implementación	Implementada

### 3.14 TARJETA CRC NO 14: EDITAR UBICACIONES

**Tabla 58: Tarjeta CRC No 14: Editar Ubicaciones**

TARJETA CRC No. 14	
Nombre de la Clase	Ubicación Controller
Nombre de la Función	Editar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 014
Resumen	
Modificar la ubicación creada.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Identificar la ubicación y código de ubicación a modificar.</li> <li>➤ ingresar a la opción modificar, estableciendo código de ubicación a editar.</li> <li>➤ Verificar que la ubicación a modificar sea la proporcionada por el sistema.</li> <li>➤ Editar campos de información, guardar los cambios realizados.</li> </ul>	
Colaboraciones	
<b>Ubicación</b> Model: Public Function <b>Editar</b> Ubicacion	
Views: Editar	
Estado de implementación	Implementada

### 3.15 TARJETA CRC NO 15: ELIMINAR UBICACIONES

**Tabla 59: Tarjeta CRC No 15: Eliminar Ubicaciones**

TARJETA CRC No. 15	
Nombre de la Clase	UbicacionController
Nombre de la Función	Eliminar
Módulo del Sistema	Manejo y Control



Código de la Tarjeta CRC	CRC- MC- 015
Resumen	
Eliminar ubicación previamente habilitada.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Identificar ubicación.</li> <li>➤ Listar ubicación a partir del código.</li> <li>➤ Elegir la opción de eliminar.</li> <li>➤ Guardar cambios.</li> <li>➤ Validar la información.</li> <li>➤ Verificar la eliminación de la ubicación.</li> </ul>	
Colaboraciones	
<b>Ubicacion</b> Model: Public Function EliminarUbicacion	
Views: Eliminar	
Estado de implementación	Implementada

### 3.16 TARJETA CRC NO 16: LISTAR UBICACIONES

**Tabla 60: Tarjeta CRC No 16: Listar Ubicaciones**

<b>TARJETA CRC No. 16</b>	
Nombre de la Clase	UbicacionController
Nombre de la Función	Listar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 016
Resumen	
Mostrar ubicaciones ingresadas en el sistema.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Ingresar al módulo de manejo y control.</li> <li>➤ Listar ubicación del sistema.</li> <li>➤ Verificar listado de ubicaciones.</li> </ul>	
Colaboraciones	
<b>Ubicacion</b> Model: Public Function GetUbicacionAll	
Views: Listar	
Estado de implementación	Implementada

### 3.17 TARJETA CRC NO 17: CONSULTAR UBICACIONES

**Tabla 61: Tarjeta CRC No 17: Consultar Ubicaciones**

TARJETA CRC No. 17	
Nombre de la Clase	UbicacionController
Nombre de la Función	ListarAjax
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 017
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Determinar criterio a consultar en ubicación.</li> <li>➤ Ingresar información.</li> <li>➤ Validar información</li> <li>➤ Informar Resultado.</li> </ul>	
Colaboraciones	
Ubicacion Model: Public Function GetUbicacionAll Views: ListarAjax	
Estado de implementación	Implementada

### 3.18 TARJETA CRC NO 18: AGREGAR PADRINO

**Tabla 62: Tarjeta CRC No 18: Agregar Padrino**

TARJETA CRC No. 18	
Nombre de la Clase	PadrinoController
Nombre de la Función	Nuevo
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 018
Resumen	
Crear nuevo padrino	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos personales del padrino a crear.</li> <li>➤ Ingresar información</li> <li>➤ Guardar información ingresada.</li> <li>➤ Validar información</li> <li>➤ Informar el resultado de la operación.</li> </ul>	
Colaboraciones	
PadrinoModel: Public Function InsertarPadrino Views: Nuevo	
Estado de implementación	Implementada

### 3.19 TARJETA CRC NO 19: EDITAR PADRINO

**Tabla 63: Tarjeta CRC No 19: Editar Padrino**

TARJETA CRC No. 19	
Nombre de la Clase	ACLController
Nombre de la Función	Editar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC-MC- 019
Resumen	
Modificar la información del padrino creado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información de padrino a modificar.</li> <li>➤ Ingresar código de padrino, verificar datos del padrino.</li> <li>➤ Ingresar a editar y generar cambios en la información del padrino.</li> <li>➤ Validar información.</li> <li>➤ Informar resultado de la operación.</li> </ul>	
Colaboraciones	
ACL Model: Public Function EditarPadrino	
Views: Editar	
Estado de implementación	Implementada

### 3.20 TARJETA CRC NO 20: ELIMINAR PADRINO

**Tabla 64: Tarjeta CRC No 20: Eliminar Padrino**

TARJETA CRC No. 20	
Nombre de la Clase	PadrinoController
Nombre de la Función	Eliminar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 020
65: Resumen	
Eliminar Padrino Creado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información del padrino a eliminar.</li> <li>➤ Ingresar al módulo de manejo y control, padrinos modificar o eliminar.</li> <li>➤ Ingresar código del padrino y seleccionar</li> <li>➤ Validar información.</li> <li>➤ Eliminar padrino.</li> <li>➤ Informar resultado de la operación.</li> </ul>	
Colaboraciones	

ACL Model: Public Function EliminarPadrino	
Views: Eliminar	
Estado de implementación	Implementada

### 3.21 TARJETA CRC NO 21: LISTAR PADRINO

**Tabla65 Tarjeta CRC No 21: Listar Padrino**

TARJETA CRC No. 21	
Nombre de la Clase	PadrinoController
Nombre de la Función	Listar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 021
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Ingresar al módulo de manejo y control.</li> <li>➤ Listar padrino del sistema.</li> <li>➤ Verificar listado de padrinos.</li> </ul>	
Colaboraciones	
PadrinoModel: Public Function GetPadrinos	
Views: Listar	
Estado de implementación	Implementada

### 3.22 TARJETA CRC NO 22: CONSULTAR PADRINOS

**Tabla67 Tarjeta CRC No 22: Consultar Padrinos**

TARJETA CRC No. 22	
Nombre de la Clase	PadrinosController
Nombre de la Función	ListarAjax
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 022
Resumen	
Modificar la información del padrino registrado.	
Responsabilidades	

<ul style="list-style-type: none"> <li>➤ Módulo de manejo y control, listar padrinos en consulta.</li> <li>➤ Solicitar datos de padrino específico a consultar.</li> <li>➤ Verificar datos personales del padrino, estado, y fecha de actualización.</li> </ul>	
Colaboraciones	
PadrinosModel: Public Function GetPadrinos	
Views: ListarAjax	
Estado de implementación	Implementada

### 3.23 TARJETA CRC NO 23: AGREGAR INFORMACIÓN NIÑO

**Tabla68 Tarjeta CRC No 23: Agregar Información Niño**

<b>TARJETA CRC No. 23</b>	
Nombre de la Clase	IndexController
Nombre de la Función	Nuevo
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 023
Resumen	
Ingresar nuevo ahijado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos personales el niño y permiso para la creación del nuevo niño.</li> <li>➤ Validar la información para el registro del niño.</li> <li>➤ Ingresar a la base de datos el nuevo niño</li> <li>➤ Informar el resultado de la operación</li> </ul>	
Colaboraciones	
IndexModel: Public Function GetCrio	
Views: Nuevo	
Estado de implementación	Implementada

### 3.24 TARJETA CRC NO 24: EDITAR INFORMACIÓN NIÑO

**Tabla69 Tarjeta CRC No 24: Editar Información Niño**

<b>TARJETA CRC No. 24</b>	
Nombre de la Clase	FamiliaController
Nombre de la Función	Editar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 024
Resumen	
Modificar la información del niño ingresado	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información del niño a realizar modificaciones.</li> <li>➤ Listar niños ingresados en el sistema.</li> </ul>	

<ul style="list-style-type: none"> <li>➤ Ingresar a editar y modificar los datos solicitados.</li> <li>➤ Guardar cambios.</li> <li>➤ Validar la información</li> <li>➤ Informar procesos realizados.</li> </ul>	
Colaboraciones	
Familia Model: Public Function GetFamilia	
Views: Editar	
Estado de implementación	Implementada

### 3.25 TARJETA CRC NO 25: ELIMINAR INFORMACIÓN NIÑO

**Tabla70 Tarjeta CRC No 25: Eliminar Información Niño**

TARJETA CRC No. 25	
Nombre de la Clase	Familia Controller
Nombre de la Función	Eliminar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 025
Resumen	
Eliminar niño ingresado al sistema.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos del niño a eliminar.</li> <li>➤ Consultar niños del sistema de información, listar.</li> <li>➤ Ingresar código de niño, validar la información.</li> <li>➤ Seleccionar la opción de eliminar niño.</li> <li>➤ Validar la información.</li> <li>➤ Informar resultado del proceso.</li> </ul>	
Colaboraciones	
Familia Model: Public Function EliminarFamilia	
Views: Eliminar	
Estado de implementación	Implementada

### 3.26 TARJETA CRC NO 26: LISTAR INFORMACIÓN NIÑO

**Tabla71 Tarjeta CRC No 26: Listar Información Niño**

TARJETA CRC No. 26	
Nombre de la Clase	Familia Controller
Nombre de la Función	Listar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 026
Resumen	
Muestra todos los niños ingresados al sistema y sus datos personales.	

Responsabilidades	
<ul style="list-style-type: none"> <li>➤ En el módulo de Manejo y Control, en ahijados opción listar, permitirá identificar todos los niños ingresados en el sistema.</li> <li>➤ Puede acceder a editar o eliminar según corresponda.</li> </ul>	
Colaboraciones	
Familia Model: Public Function GetFamiliasAll	
Views: Listar	
Estado de implementación	Implementada

### 3.27 TARJETA CRC NO 27: CONSULTAR INFORMACIÓN NIÑOS

**Tabla72 Tarjeta CRC No 27: Consultar Información Niños**

TARJETA CRC No. 27	
Nombre de la Clase	IndexController
Nombre de la Función	IndexAjax
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 027
Resumen	
Permite conocer y/o verificar información de los niños ingresados al sistema.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Determinar criterio a consultar en ahijados.</li> <li>➤ Ingresar información.</li> <li>➤ Validar información</li> <li>➤ Informar Resultado.</li> </ul>	
Colaboraciones	
Index Model: Public Function GetDatosCrio	
Views: IndexAjax	
Estado de implementación	Implementada

### 3.28 TARJETA CRC NO 28: CREAR APADRINAMIENTO

**Tabla73 Tarjeta CRC No 28: Crear Apadrinamiento**

TARJETA CRC No. 28	
Nombre de la Clase	ApadrinamientoController
Nombre de la Función	Nuevo
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 028
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	

<ul style="list-style-type: none"> <li>➤ Solicitar datos del proceso de apadrinamiento a crear.</li> <li>➤ Ingresar información</li> <li>➤ Guardar información ingresada.</li> <li>➤ Validar información</li> <li>➤ Informar el resultado de la operación.</li> </ul>	
Colaboraciones	
Apadrinamiento ACL Model: Public Function GetApadrinamiento	
Views: Nuevo	
Estado de implementación	Implementada

### 3.29 TARJETA CRC NO 29: EDITAR APADRINAMIENTO

**Tabla74 Tarjeta CRC No 29: Editar Apadrinamiento**

TARJETA CRC No. 29	
Nombre de la Clase	ApadrinamientoController
Nombre de la Función	Editar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 029
Resumen	
Modificar la información del proceso de apadrinamiento ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información del proceso de apadrinamiento a modificar.</li> <li>➤ Listar proceso de apadrinamiento en el sistema.</li> <li>➤ Ingresar a editar y modificar los datos solicitados.</li> <li>➤ Guardar cambios.</li> <li>➤ Validar la información</li> <li>➤ Informar procesos realizados.</li> </ul>	
Colaboraciones	
Apadrinamiento Model: Public Function EditarApadrinamiento	
Views: Editar	
Estado de implementación	Implementada

### 3.30 TARJETA CRC NO 30: ELIMINAR APADRINAMIENTO

**Tabla75 Tarjeta CRC No 30: Eliminar Apadrinamiento**

TARJETA CRC No. 30	
Nombre de la Clase	ApadrinamientoController
Nombre de la Función	Eliminar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- AU- 002
Resumen	
Modificar la información del usuario ingresado.	



Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos del proceso de apadrinamiento a eliminar.</li> <li>➤ Consultar proceso de apadrinamiento del sistema de información, listar.</li> <li>➤ Ingresar código de apadrinamiento.</li> <li>➤ Validar la información.</li> <li>➤ Seleccionar la opción de eliminar.</li> <li>➤ Validar la información.</li> <li>➤ Informar resultado del proceso.</li> </ul>	
Colaboraciones	
Apadrinamiento Model: Public Function EliminarApadrinamiento	
Views: Eliminar	
Estado de implementación	Implementada

### 3.31 TARJETA CRC NO 31: LISTAR APADRINAMIENTO

**Tabla76 Tarjeta CRC No 31: Listar Apadrinamiento**

TARJETA CRC No. 31	
Nombre de la Clase	ApadrinamientoController
Nombre de la Función	Listar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 031
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Desde el módulo de manejo y control, listar proceso apadrinar.</li> <li>➤ Identificar listado de niños ingresados en el proceso de apadrinamiento.</li> <li>➤ Visualizar opciones de editar y eliminar, según corresponda</li> <li>➤ Evaluar información contenida.</li> </ul>	
Colaboraciones	
Apadrinamiento Model: Public Function GetApadrinamientoAll	
Views: Listar	
Estado de implementación	Implementada

### 3.32 TARJETA CRC NO 32: CONSULTAR APADRINAMIENTO

**Tabla77 Tarjeta CRC No 32: Consultar Apadrinamiento**

TARJETA CRC No. 32	
Nombre de la Clase	ApadrinamientoController
Nombre de la Función	IndexAjax
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 032
Resumen	
Modificar la información del usuario ingresado.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Determinar criterio a consultar en Apadrinamiento</li> <li>➤ Ingresar información.</li> <li>➤ Validar información</li> <li>➤ Informar Resultado.</li> </ul>	
Colaboraciones	
Apadrinamiento Model: Public Function GetApadrinamientoAll Views: Listar	
Estado de implementación	Implementada

### 3.33 TARJETA CRC NO 33: AGREGAR DIBUJO

**Tabla78 Tarjeta CRC No 33: Agregar Dibujo**

TARJETA CRC No. 33	
Nombre de la Clase	CampDibujoController
Nombre de la Función	Nuevo
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 031
Resumen	
Ingresar nuevo dibujo.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos del dibujo a ingresar, código del niño y de la campaña.</li> <li>➤ Ingresar información, estado del dibujo.</li> <li>➤ Guardar cambios</li> <li>➤ Validar información.</li> <li>➤ Informar resultado del proceso.</li> </ul>	
Colaboraciones	
CampDibujo Model: Public Function InsertarDibujo Views:Nuevo	
Estado de implementación	Implementada

### 3.34 TARJETA CRC NO 34: EDITAR DIBUJO

**Tabla79 Tarjeta CRC No 34: Editar Dibujo**

TARJETA CRC No. 34	
Nombre de la Clase	CampDibujoController
Nombre de la Función	Editar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 034
Resumen	
Modificar los dibujos ingresados al sistema.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar información del dibujo a modificar.</li> <li>➤ Listar proceso de dibujo en el sistema, o directamente ir por modificar.</li> <li>➤ Ingresar a editar y modificar los datos solicitados.</li> <li>➤ Guardar cambios.</li> <li>➤ Validar la información</li> <li>➤ Informar procesos realizados.</li> </ul>	
Colaboraciones	
CampDibujo Model: Public Function EditarDibujo Views:Editar	
Estado de implementación	Implementada

### 3.35 TARJETA CRC NO 35: ELIMINAR DIBUJO

**Tabla80 Tarjeta CRC No 35: Eliminar Dibujo**

TARJETA CRC No. 35	
Nombre de la Clase	CampDibujoController
Nombre de la Función	Eliminar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 035
Resumen	
Modificar los dibujos ingresados al sistema.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Solicitar datos Del dibujo a eliminar.</li> <li>➤ Consultar proceso de dibujo en el sistema de información, listar.</li> <li>➤ Ingresar código del dibujo.</li> <li>➤ Validar la información.</li> </ul>	

<ul style="list-style-type: none"> <li>➤ Seleccionar la opción de eliminar.</li> <li>➤ Validar la información.</li> <li>➤ Informar resultado del proceso.</li> </ul>	
Colaboraciones	
CampDibujo Model: Public Function EliminarDibujo Views: Eliminar	
Estado de implementación	Implementada

### 3.36 TARJETA CRC NO 36: LISTAR DIBUJO

**Tabla81 Tarjeta CRC No 36: Listar Dibujo**

TARJETA CRC No. 36	
Nombre de la Clase	CampDibujoController
Nombre de la Función	Listar
Módulo del Sistema	Manejo y Control
Código de la Tarjeta CRC	CRC- MC- 036
Resumen	
Permite visualizar los dibujos ingresados al sistema por campaña y niño.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Ingresar a través del módulo de manejo y control, campañas, listar.</li> <li>➤ Identificar número de campaña y nombre de niños con dibujos ingresados.</li> <li>➤ Visualizar información de cada dibujo por ahijado.</li> <li>➤ Ingresar información de dibujo por ahijado en caso de requerir más específica la información.</li> </ul>	
Colaboraciones	
CampDibujo Model: Public Function GetDibujo Views: Listar	
Estado de implementación	Implementada

### 3.37 TARJETA CRC NO 37: CONSULTAR DIBUJO

**Tabla82 Tarjeta CRC No 37: Consultar Dibujo**

TARJETA CRC No. 37	
Nombre de la Clase	CampDibujoController
Nombre de la Función	ListarAjax
Módulo del Sistema	Manejo y Control

Código de la Tarjeta CRC	CRC- MC- 037
Resumen	
Permite visualizar los dibujos ingresados al sistema por campaña y niño.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ Determinar criterio a consultar en dibujo.</li> <li>➤ Ingresar información.</li> <li>➤ Validar información</li> <li>➤ Informar Resultado.</li> </ul>	
Colaboraciones	
CampDibujo Model: Public Function GetDibujo Views: ListarAjax	
Estado de implementación	Implementada

### 3.38 TARJETA CRC NO 38: EXPORTAR DATOS A ESPAÑA

**Tabla83 Tarjeta CRC No 38: Exportar Datos a España**

<b>TARJETA CRC No. 38</b>	
Nombre de la Clase	ExportarController
Nombre de la Función	EspañaAll
Módulo del Sistema	Exportar
Código de la Tarjeta CRC	CRC- RI- 038
Resumen	
Enviar las bases de datos con la información completa del proceso de apadrinamiento.	
Responsabilidades	
<ul style="list-style-type: none"> <li>➤ En el módulo exportar, ingresar a bases a España.</li> <li>➤ Determinar tipo de base de datos a exportar, sin bajas, con baja o completas.</li> <li>➤ Validar la información.</li> <li>➤ Visualizar la base de datos.</li> </ul>	
Colaboraciones	
Exportar Model: Public Function GetCrios Views: BaseEspana	
Estado de implementación	Implementada

## 4. FASE DE DESARROLLO: UNIDADES DE PRUEBA

En esta fase se realizan las pruebas a la fase anterior, es aquí donde se verifica que realmente el sistema de información cumpla con lo planeado, sirve para identificar fallas o detectar información no necesaria o sin importancia. Se relaciona más con el código fuente de programación y generamos una tabla por cada clase o módulo programado.

### 4.1 UNIDAD DE PRUEBA NO1: ACLCONTROLLER

**Tabla84 Unidad de prueba No1: ACLController**

UNIDAD DE PRUEBA No. 1	
<b>Nombre de la Clase</b>	ACLController
<b>Modulo del Sistema</b>	Administración/Usuarios
<b>Código de la Tarjeta CRC</b>	CRC-AU-001, CRC-AU-002, CRC-AU-003, CRC-AU-004, CRC-AU-005, CRC-AU-006
<b>Código de la Prueba</b>	UPR-AU-001
¿Cumple la clase con los siguientes requisitos?	
La vista relacionada con la clase, implementa un formulario para consultar, ingresar, modificar o eliminar información de la base de datos?	Si
Es amigable la interface de usuario?	Si
Valida si el usuario tiene permiso para realizar actividades en esta clase?	Si
Valida los datos ingresados para prevenir errores de inserción, duplicidades, etc.?	Si
Muestra información al usuario sobre el resultado de la transacción?	Si
Instancia y relaciona la clase con la Vista respectiva e interfaces correspondientes a los CRC relacionados en esta prueba?	Si
La clase implementa validación de datos en caso de no usar GOOGLE CHROME?	Si
Está implementado el método para crear nuevos roles y permisos	Si
Está implementado el método para editar la información de roles y permisos	Si
Está implementado el método para eliminar roles y permisos	Si
Está implementado el método para asignarle permisos al role ingresado	Si
Los métodos cuentan con un interface o vista para interactuar con el usuario	Si
<b>Calificación a la prueba</b>	Implementada correctamente la clase

## 4.2 UNIDAD DE PRUEBA NO 2: REGISTROCONTROLLER

**Tabla85 Unidad de prueba No 2: RegistroController**

UNIDAD DE PRUEBA No. 2	
<b>Nombre de la Clase</b>	RegistroController
<b>Modulo del Sistema</b>	Administración/Usuarios
<b>Código de la Tarjeta CRC</b>	CRC- AU- 007, CRC- AU- 008, CRC- AU- 010,CRC- AU- 011
<b>Código de la Prueba</b>	UPR-AU-002
<b>¿Cumple la clase con los siguientes requisitos?</b>	
La vista relacionada con la clase, implementa un formulario para consultar, ingresar, modificar o eliminar información de la base de datos?	Si
Es amigable la interface de usuario?	Si
Valida si el usuario tiene permiso para realizar actividades en esta clase?	Si
Valida los datos ingresados para prevenir errores de inserción, duplicidades, etc.?	Si
El formulario permite la validación de datos antes de guardar?	Si
Muestra información al usuario sobre el resultado de la transacción?	Si
Instancia y relaciona la clase con la Vista respectiva e interfaces correspondientes a los CRC relacionados en esta prueba?	Si
La clase implementa validación de datos en caso de no usar GOOGLE CHROME?	Si
Está implementado el método para crear nuevos usuarios	Si
Está implementado el método para editar la información de usuarios	Si
Está implementado el método para eliminar usuarios	Si
Está implementado el método para asignarle los permisos correspondientes a su role	Si
<b>Calificación a la prueba</b>	Implementada correctamente la clase

### 4.3 UNIDAD DE PRUEBA NO 3: USUARIOS-INDEXCONTROLLER

**Tabla86 Unidad de prueba No 3: Usuarios-indexController**

UNIDAD DE PRUEBA No. 3	
<b>Nombre de la Clase</b>	Usuarios-indexController
<b>Módulo del Sistema</b>	Administración/Usuarios
<b>Código de la Tarjeta CRC</b>	CRC- AU- 003
<b>Código de la Prueba</b>	UPR-AU-009
¿Cumple la clase con los siguientes requisitos?	
La vista relacionada con la clase, implementa un formulario para configurar los permisos del usuarios?	Si
Es amigable la interface de usuario?	Si
Valida si el usuario tiene permiso para configurar permisos?	Si
El formulario permite la validación de datos antes de guardar?	Si
Muestra información al usuario sobre el resultado de la transacción?	Si
Implementa una vista para la configuración de usuarios, como son: el cambio de contraseña, modificar los datos del usuario, activar o desactivar la cuenta y cambiar los permisos que tiene el usuario en el sistema de información	Si
<b>Calificación a la prueba</b>	Implementada correctamente la clase



## 5. FASES DE PRUEBA DE ACEPTACION

Esta fase es la prueba de que todo funciona tal cual nos contaron los usuarios en las historias, realizan las pruebas y verifican los caminos por los cuales pueden ser desviados, es una especie de validaciones, donde identificamos si funciona o nos hace falta algo en el sistema.

### 5.1 PRUEBAS DE ACEPTACIÓN NO.1: INICIAR LA APLICACIÓN

Tabla 87 Pruebas de aceptación No.1: Iniciar la aplicación

<b>Prueba de Aceptación No 1</b>	
<b>Título de la prueba</b>	Iniciar la aplicación
<b>Módulo del sistema</b>	General
<b>Códigos de las historias</b>	HU – GR- 001, HU – GR- 002, HU – GR- 003
<b>Código de la prueba</b>	PRA- GR- 001
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	La aplicación podrá iniciarse desde cualquier equipo que cuente con conexión online, acceder por la página de la fundación y esperar a que este se conecte y aparezca la ventana iniciar sesión.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"><li>☞ La red de la entidad o del usuario deberá permitir la conexión desde internet al sistema.</li><li>☞ El sistema requiere el explorador Google Chrome para un mejor funcionamiento.</li><li>☞ Cada usuario deberá contar con una identificación y contraseña para acceder al sistema.</li></ul>
<b>Curso de la ejecución</b>	

<b>Camino No 1</b>	<p>Curso Normal de la ejecución:</p> <ul style="list-style-type: none"> <li>→ Acceder al explorador Google Chrome.</li> <li>→ Una vez abierto el explorador, introducir el link: <a href="https://apps.globalhumanitariacolombia.org/siglha/">apps.globalhumanitariacolombia.org/siglha/</a></li> <li>→ Dar enter.</li> <li>→ Aparecerá la interfaz del sistema de información.</li> <li>→ Dar click en el botón iniciar sesión ubicado en la parte superior derecha de la ventana.</li> <li>→ Ingrese su usuario y contraseña en el campo habilitado para esto.</li> <li>→ Dar click en entrar.</li> </ul>
<b>Camino No 2</b>	<p><u>No puede conectarse</u>  Verifique la conexión de internet, y vuelva a intentarlo.</p>
<b>Camino No 3</b>	<p><u>Not Found</u></p> <ul style="list-style-type: none"> <li>→ Verifique el link utilizado para acceder a la interfaz.</li> <li>→ Copie el siguiente link: <a href="https://apps.globalhumanitariacolombia.org/siglha/">apps.globalhumanitariacolombia.org/siglha/</a></li> <li>→ De Enter.</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<ul style="list-style-type: none"> <li>→ Deberá aparecer la interfaz principal del sistema de información SIGLHA.</li> <li>→ Debe visualizarse el botón iniciar sesión, ubicado en la parte superior mano derecha de la ventana.</li> </ul>
<b>Camino No 2</b>	<p><u>No puede conectarse</u></p> <ul style="list-style-type: none"> <li>→ Se introdujo el link del sistema de información.</li> <li>→ Aparece un mensaje de alerta indicando que no se puede conectar.</li> <li>→ Se verifica la conexión, se vuelve a conectar a internet.</li> <li>→ Se da nuevamente Enter sobre el link.</li> <li>→ La aplicación inicio y mostro la interfaz principal del sistema.</li> </ul>

<b>Camino No 3</b>	<u>Not Found</u> → Se introdujo el link del sistema de información. → Aparece un mensaje de alerta indicando que fallo Not Found. → Se verifica el link, se observa error. → Se copia nuevamente el link, enter. → La aplicación inicio y mostro la interfaz principal del sistema.
<b>Calificación de la prueba</b>	Las historias se han implementado correctamente.

## 5.2 PRUEBAS DE ACEPTACIÓN NO. 2: INICIAR SESIÓN

**Tabla88 Pruebas de aceptación No. 2: Iniciar sesión**

<b>Prueba de Aceptación No 2</b>	
<b>Título de la prueba</b>	Iniciar Sesión
<b>Módulo del sistema</b>	General
<b>Códigos de las historias</b>	HU – GR- 001, HU – GR- 002, HU – GR- 003
<b>Código de la prueba</b>	PRA- GR- 002
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Iniciar la sesión de aplicación a partir del usuario y role generado, así como los permisos establecido para el cumplimiento de sus funciones, otorgados por el administrador.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA-GR-001 (o el camino 2 o 3 si las operaciones no resultan exitosas).</li> <li>☞ Debe estar visible la pantalla de inicio de sesión.</li> <li>☞ El usuario deberá contar con un nombre de usuario y contraseña previamente otorgada.</li> </ul>
<b>Curso de la ejecución</b>	

<b>Camino No 1</b>	<u>Nombre de usuario y password correcto:</u> → Dar click en el botón iniciar sesión ubicado en la parte superior derecha de la ventana. → Ingrese su usuario y password en el campo habilitado para esto. → Dar click en entrar.
<b>Camino No 2</b>	<u>Usuario y /o Password incorrecto</u> → Verifique el usuario y password. → Vuelva a intentar ( <i>Repita pasos Camino No 1</i> ) → En caso de volver a presentar el inconveniente y no poder acceder, comunicarse con el administrador.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Nombre de usuario y password correcto:</u> → El sistema deberá verificar en sus bases de datos la información del usuario. → El sistema compara la información suministrada con la información almacenada. → Validar la información y encontrar coincidencias. → El sistema deberá entregar un mensaje que diga “Bienvenido xxxx” → El usuario podrá acceder a la aplicación, según permisos obtenidos.
<b>Camino No 2</b>	<u>Usuario y /o Password incorrecto</u> → El sistema deberá buscar en la base de datos la información de los usuarios. → Comparar la información suministrada con la almacenada. → La ventana de inicio no se cierra y debe mostrar un mensaje notificando el error de usuario o password.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Nombre de usuario y password correcto:</u> → El sistema genera el mensaje de

	<p>bienvenida al usuario.</p> <p>→ Aparece la interfaz principal la cual queda a disposición del usuario.</p>
<b>Camino No 2</b>	<p><u>Usuario y /o Password incorrecto</u></p> <p>→ Aparece el mensaje de usuario o password incorrecto.</p> <p>→ El sistema solicita nuevamente los datos de usuario y password.</p>
<b>Calificación Obtenida</b>	Las Historias han sido implementadas correctamente.

### 5.3 PRUEBAS DE ACEPTACIÓN NO. 3: INGRESAR NUEVO USUARIO

**Tabla89 Pruebas de aceptación No. 3: Ingresar Nuevo Usuario**

<b>Prueba de Aceptación No 3</b>	
<b>Título de la prueba</b>	Nuevo Usuario
<b>Módulo del sistema</b>	Administrador Usuario
<b>Códigos de las historias</b>	HU- AU- 011
<b>Código de la prueba</b>	PRA- AU- 003
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Ingresar los datos que identifican a un nuevo usuario, indicando su usuario, contraseña role y estado, partiendo de que los Roles ya cuentan con los permisos establecidos.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA-GR-001 (o el camino 2 o 3 si las operaciones resultan exitosas), y el Camino No1 del PRA-GR-002 (o el camino 2 si las operaciones resultan exitosas),</li> <li>☞ Debe estar visible el formulario de registro de nuevos usuarios.</li> <li>☞ Contar con los permisos para el registro de nuevos usuarios.</li> </ul>
<b>Curso de la ejecución</b>	

<p><b>Camino No 1</b></p>	<p><u>Registro de usuario Exitoso</u></p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de administración/usuario.</li> <li>→ Dar click en la opción usuario y luego en la opción ingresar.</li> <li>→ Deberá aparecer el formulario a diligenciar para la creación del nuevo usuario.</li> <li>→ Ingresar información solicitada.</li> <li>→ Definir role y estado.</li> <li>→ Dar click en enviar.</li> <li>→ El sistema generar el mensaje de usuario registrado satisfactoriamente.</li> </ul>
<p><b>Camino No 2</b></p>	<p><u>Campos obligatorios faltantes</u></p> <ul style="list-style-type: none"> <li>→ El sistema generara el mensaje de <i>Debe introducir x campo</i>, según corresponda (Primer nombre, primer apellido, usuario, email institucional, password, confirmar, role, estado).</li> <li>→ El sistema solicitara la información faltante.</li> </ul>
<p><b>Camino No 3</b></p>	<p><u>Usuario ya registrado</u></p> <ul style="list-style-type: none"> <li>→ El sistema arrojará el mensaje de que el usuario ya existe.</li> <li>→ El sistema solicitara modificar la información.</li> </ul>
<p><b>Resultados esperados</b></p>	
<p><b>Camino No 1</b></p>	<p><u>Registro de usuario Exitoso</u></p> <ul style="list-style-type: none"> <li>→ El sistema deberá verificar en sus bases de datos la información del usuario.</li> <li>→ El sistema compara las características de la información suministrada</li> <li>→ Validar la información.</li> <li>→ El sistema deberá entregar un mensaje que diga “Usuario Registrado satisfactoriamente”</li> <li>→ El usuario podrá acceder a la aplicación, según permisos obtenidos.</li> </ul>

<b>Camino No 2</b>	<u>Campos obligatorios faltantes</u> → El sistema verificara el diligenciamiento completo del formulario, principalmente de los campos obligatorios. → El sistema generara el mensaje de <i>Debe introducir x campo</i> , según corresponda (Primer nombre, primer apellido, usuario, email institucional, password, confirmar, role, estado). → El sistema solicitara la información faltante.
<b>Camino No 3</b>	<u>Usuario ya registrado</u> → El sistema verificara el diligenciamiento completo y adecuado del formulario de creación. → Verificara la existencia de coincidencia de la información ingresada con información existente en el sistema. → El sistema arrojara el mensaje de que el usuario ya existe.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro de usuario Exitoso</u> → El sistema valida la información y la acepta. → El sistema entrega el mensaje de “Usuario Registrado satisfactoriamente”
<b>Camino No 2</b>	<u>Campos obligatorios faltantes</u> → Aparece el mensaje de campo obligatorio faltante. → El sistema solicita diligenciar completamente el formulario nuevamente los datos de usuario y password. → El sistema entrega el mensaje de “Usuario Registrado satisfactoriamente”
<b>Camino No 3</b>	<u>Usuario ya registrado</u> → El sistema arrojara el mensaje de que el usuario ya existe. → El sistema solicitar verificar,

	modificar o eliminar la información.
<b>Calificación Obtenida</b>	Historias Habilitadas satisfactoriamente

#### 5.4 PRUEBAS DE ACEPTACIÓN NO. 4: EDITAR USUARIO

Tabla90 Pruebas de aceptación No. 4: Editar Usuario

<b>Prueba de Aceptación No 4</b>	
<b>Título de la prueba</b>	Editar Datos Usuario
<b>Módulo del sistema</b>	Administrador Usuario
<b>Códigos de las historias</b>	HU- AU- 011, HU- AU- 012
<b>Código de la prueba</b>	PRA- AU- 004
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Modificar datos de un usuario previamente creado.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA-AU-003 (o el camino 2 o 3 si las operaciones no resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → Dar click en el módulo de administración/usuario. → Dar click en la opción usuario y luego en la opción editar. → Deberá aparecer listados los usuarios con sus respectivos roles, estados. → Cuenta con la opción editar permisos y datos de la cuenta, y cambiar contraseña. → Dar click sobre el elemento a modificar. → Realizar cambios. → Dar click en guardar. → El sistema arrojará el mensaje de <i>Registro Modificado</i> . →



<b>Camino No 2</b>	<u>Eliminación de un campo obligatorio</u> → El sistema verificara el diligenciamiento completo de los campos, principalmente de los campos obligatorios. → Validara la información. → El sistema generara el mensaje de <i>Debe introducir x campo</i> , según corresponda (Usuario, email institucional, password, confirmar, role, estado).
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Registro Modificado Exitosamente</u> → El sistema deberá verificar la información modificada del usuario. → Validar la información. → El sistema deberá entregar un mensaje que diga “Registro Modificado”
<b>Camino No 2</b>	<u>Eliminación de un campo obligatorio</u> → El sistema verificara el diligenciamiento completo de los campos, principalmente de los campos obligatorios. → Validara la información. → El sistema generara el mensaje de <i>Debe introducir x campo</i> , según corresponda (Usuario, email institucional, password, confirmar, role, estado) → El sistema solicitara la información faltante.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro modificado Exitosamente</u> → El sistema valida la información y la acepta. → El sistema entrega el mensaje de “Registro Modificado”.
<b>Camino No 2</b>	<u>Eliminación de un campo obligatorio</u> → Aparece el mensaje de campo obligatorio faltante. → El sistema solicita diligenciar información faltante. → Se cierra el formulario.

Calificación Obtenida	Historias Habilitadas satisfactoriamente
-----------------------	--

## 5.5 PRUEBA DE ACEPTACIÓN NO 5: CREAR UN ROLE

Tabla91 Prueba de Aceptación No 5: Crear un Role

Prueba de Aceptación No 5	
Título de la prueba	Crear un Nuevo Role
Módulo del sistema	Administrador Usuario
Códigos de las historias	HU- AU- 005
Código de la prueba	PRA- AU- 005
Datos del caso de la prueba	
Descripción de la prueba	Ingresar role y permisos al nuevo role.
Condiciones de la ejecución	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA-GR-001 (o el camino 2 o 3 si las operaciones resultan exitosas), y el Camino No1 del PRA-GR-002 (o el camino 2 si las operaciones resultan exitosas),</li> <li>☞ Debe estar visible el formulario de registro de nuevos usuarios.</li> <li>☞ Contar con los permisos para el registro de nuevo Role.</li> </ul>
Curso de la ejecución	
Camino No 1	<u>Registro de Role Exitoso</u> → Dar click en el módulo de administración/usuario. → Dar click en la opción lista de control de cambios → Deberá aparecer las opciones Roles y Permisos. → Dar click en Roles. → Se listan los Roles y aparecerá a opción nuevo Role. → Definir role. → Dar click en Guardar.
Camino No 3	<u>Role ya registrado</u> → El sistema arrojará el mensaje de que el Role ya existe. → El sistema solicitará modificar la

	información.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Registro de Role Exitoso</u> → El sistema deberá verificar en sus bases de datos el nuevo Role Ingresado. → El sistema evaluara la existencia del Role. → Validar la información. → El sistema Cerrara el formulario de nuevo Role, y generara la interfaz de listar Roles.
<b>Camino No 3</b>	<u>Role ya registrado</u> → El sistema deberá verificar en sus bases de datos el nuevo Role Ingresado. → Verificara la existencia de coincidencia del Role ingresado. → El sistema arrojara el mensaje de que el Role ya existe.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro de Role Exitoso</u> → El sistema valida la información y la acepta. → El sistema Cierra el formulario de nuevo Role, y generara la interfaz de listar Roles.
<b>Camino No 3</b>	<u>Usuario ya registrado</u> → El sistema arrojara el mensaje de que el Role ya existe. → El sistema solicitar verificar, modificar o eliminar la información.
<b>Calificación Obtenida</b>	Historias Habilitadas satisfactoriamente

## 5.6 PRUEBAS DE ACEPTACIÓN NO. 6: LISTA DE CONTROL DE ACCESO ACL

Tabla 92: Pruebas de aceptación No. 6: Lista de Control de Acceso ACL

<b>Prueba de Aceptación No 6</b>	
<b>Título de la prueba</b>	Lista de Control de Acceso ACL
<b>Módulo del sistema</b>	Administrador Usuario
<b>Códigos de las historias</b>	HU – AU- 004, HU- AU- 005, HU- AU- 006 HU- AU- 007, HU- AU- 008, HU- AU- 009, HU- AU- 010.
<b>Código de la prueba</b>	PRA- AU- 006
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Visualizar el acceso a Roles y permisos.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA-AU-005 (o el camino 2 si las operaciones si resultan exitosas).</li> <li>☞ Contar con los permisos para acceder y/o modificar opciones de acceso.</li> </ul>
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Acceder a Roles: Permisos</u> → Dar click en el módulo de administración/usuario. → Dar click en la Lista de Control de Acceso. → Dar click en la opción Roles. → Se listan los roles creados. → Muestra la opción permisos y editar. → Dar click en permisos → Se muestran los permisos habilitados, denegados e ignorados. → Modificar los permisos según corresponda. → Dar click en guardar. → El sistema arrojará el Mensaje <i>Proceso Completado</i> .
<b>Camino No 2</b>	<u>Acceder a Roles: Editar</u> → Dar click en el módulo de administración/usuario. → Dar click en la Lista de Control de Acceso. → Dar click en la opción Roles.

	<ul style="list-style-type: none"> <li>→ Se listan los roles creados.</li> <li>→ Muestra la opción permisos y editar.</li> <li>→ Dar click en editar</li> <li>→ Se muestra una ventana con la opción de Role a modificar.</li> <li>→ Dar click en guardar.</li> </ul>
<b>Camino No 3</b>	<p><u>Acceder Role: Agregar Role</u></p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de administración/usuario.</li> <li>→ Dar click en la Lista de Control de Acceso.</li> <li>→ Dar click en la opción Roles.</li> <li>→ Se listan los roles creados.</li> <li>→ Muestra la opción permisos, editar.</li> <li>→ Muestra la opción agregar role.</li> <li>→ Dar click en agregar Role.</li> <li>→ Aparece la venta con los campos a diligenciar para la creación del Role.</li> <li>→ Diligenciar campo.</li> <li>→ Dar click en Guardar.</li> </ul>
<b>Camino No 4</b>	<p><u>Acceder a Permisos</u></p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de administración/usuario.</li> <li>→ Dar click en la Lista de Control de Acceso.</li> <li>→ Dar click en la opción Permisos.</li> <li>→ Se listan los permisos creados con su respectiva llave.</li> </ul>
<b>Camino No 5</b>	<p><u>Acceder Permisos: Agregar permiso</u></p> <ul style="list-style-type: none"> <li>→ Una vez listados los permisos en la parte superior izquierda de la pantalla aparecerá la opción agregar permiso.</li> <li>→ Muestra la opción agregar permiso.</li> <li>→ Dar click en agregar permisos.</li> <li>→ Diligenciar los campos de permiso con su respectiva llave.</li> <li>→ Dar click en guardar</li> <li>→ Será creado y habilitado el permiso</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<p><u>Acceder a Roles: Permisos</u></p> <ul style="list-style-type: none"> <li>→ El sistema deberá verificar la información modificada del usuario.</li> <li>→ Validar la información.</li> </ul>

	→ Guardara cambios realizados.
<b>Camino No 2</b>	<u>Acceder a Roles: Editar</u> → El sistema verificara el diligenciamiento del campo. → Validara la información. → Efectuara la modificación.
<b>Camino No 3</b>	<u>Acceder Roles: Agregar Role</u> → El sistema verificara el diligenciamiento del campo. → Validara la información. → Evaluará a existencia o no del Role solicitado. → Creará el nuevo Role.
<b>Camino No 4</b>	<u>Acceder a Permisos</u> → El sistema analizara la solicitud realizada. → Permitirá acceder al permiso.
<b>Camino No 5</b>	<u>Acceder Permisos: Agregar permiso</u> → El sistema verificara la información. → Comprobara la existencia o no del permiso.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Acceder a Roles: Permisos</u> → El sistema Valida la información y la acepta. → Guardara cambios realizados.
<b>Camino No 2</b>	<u>Acceder a Roles: Editar</u> → El sistema verificara el diligenciamiento del campo. → El sistema efectuara el cambio solicitado.
<b>Camino No 3</b>	<u>Acceder Roles: Agregar Role</u> → El sistema verificara el diligenciamiento del campo. → Crea el nuevo Role.
<b>Camino No 4</b>	<u>Acceder a Permisos</u> → El sistema analizara la solicitud realizada. → Permitirá acceder al permiso.
<b>Camino No 5</b>	<u>Acceder Permisos: Agregar permiso</u> → El sistema valido la información. → El sistema crea el permiso.

<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente
------------------------------	--

## 5.7 PRUEBAS DE ACEPTACIÓN NO. 7: LISTAR USUARIO

Tabla 93: Pruebas de aceptación No. 7: Listar Usuario

<b>Prueba de Aceptación No 7</b>	
<b>Título de la prueba</b>	Listar Usuario
<b>Módulo del sistema</b>	Administrador Usuario
<b>Códigos de las historias</b>	HU- AU- 011, HU- AU- 012, HU- AU- 013, HU- AU- 014.
<b>Código de la prueba</b>	PRA- AU- 007
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Visualizar los usuarios ingresados en el sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 003 (o el camino 2 o 3 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Usuarios</u> → Dar click en el módulo de administración/usuario. → Dar click en la editar usuario. → Deberá aparecer listados los usuarios con sus respectivos roles, estados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Usuarios</u> → El sistema validara la solicitud. → Mostrará todos los usuarios creados en el sistema.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Usuarios</u> → El sistema validara la información. → El sistema listara todos los usuarios ingresados.
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente

## 5.8 PRUEBAS DE ACEPTACIÓN NO. 8: ELIMINAR USUARIO

**Tabla 94: Pruebas de aceptación No. 8: Eliminar Usuario**

<b>Prueba de Aceptación No 8</b>	
<b>Título de la prueba</b>	Eliminar Usuario
<b>Módulo del sistema</b>	Administrador Usuario
<b>Códigos de las historias</b>	HU- AU- 011, HU- AU- 015
<b>Código de la prueba</b>	PRA- AU- 008
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Eliminar usuarios ingresados en el sistema.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA- AU- 003 (o el camino 2 si las operaciones resultan exitosas).</li> <li>☞ Contar con usuario y permisos para realizar la operación.</li> </ul>
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<p><u>Eliminar Adecuadamente el usuario</u></p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de administración/usuario.</li> <li>→ Dar click en la editar usuario.</li> <li>→ Deberá aparecer listados los usuarios con sus respectivos roles, estados, y la opción eliminar.</li> <li>→ Identificar el usuario a eliminar.</li> <li>→ Dar click en la opción eliminar del usuario a eliminar.</li> <li>→ El sistema validara la información.</li> <li>→ El sistema Mostrará una ventana con los datos del usuario solicitando confirmar la acción.</li> <li>→ Dar click en confirmar una vez verifique los datos del usuario que eliminara.</li> <li>→ El sistema eliminara el usuario.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de usuario.</li> </ul>



<b>Camino No 1</b>	<u>Eliminar usuario equivocado</u> → Dar click en el módulo de administración/usuario. → Dar click en la editar usuario. → Deberá aparecer listados los usuarios con sus respectivos roles, estados, y la opción eliminar. → Identificar el usuario a eliminar. → Dar click en la opción eliminar del usuario a eliminar. → El sistema validara la información. → El sistema Mostrará una ventana con los datos del usuario solicitando confirmar la acción. → Dar click en confirmar una vez verifique los datos del usuario que eliminara. → El sistema eliminara el usuario. → El sistema cerrara la ventana de confirmación de eliminación de usuario. → Identificar error en la eliminación
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el usuario</u> → El sistema validara la información. → El sistema eliminara el usuario.
<b>Camino No 2</b>	<u>Eliminar usuario equivocado</u> → El sistema elimina el usuario. → Se identifica error. → Crear usuario nuevamente. (Realizar pasos PRA- AU- 003)
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el usuario</u> → El sistema validara la información. → El sistema eliminara el usuario.
<b>Camino No 2</b>	<u>Eliminar usuario equivocado</u> → El sistema elimina el usuario. → Se identifica error. → Crear usuario nuevamente. (Realizar pasos PRA- AU- 003)
<b>Calificación Obtenida</b>	Historias Habilitadas Exitosamente

## 5.9 PRUEBAS DE ACEPTACIÓN NO. 9: CREAR COPIAS DE SEGURIDAD

Tabla 95: Pruebas de aceptación No. 9: Crear Copias de Seguridad

<b>Prueba de Aceptación No 9</b>	
<b>Título de la prueba</b>	Crear Copias de Seguridad
<b>Módulo del sistema</b>	Administrador Usuario
<b>Códigos de las historias</b>	HU- AU- 016, y Todos los HU del sistema.
<b>Código de la prueba</b>	PRA- AU- 009
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Generar base de datos de información para almacenar en espacios externos al online.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 de todos los PRA- del sistema (o el camino 2 0 3 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Crear BackUp Exitosamente</u> → Dar click en el módulo de administración/usuario. → Dar click en la opción copias de seguridad. → El sistema valido la información. → El sistema abre la ventana para la selección de opciones del Backus. → Seleccionar la opción a realizarle el BackUp así como el tipo de tabla deseado, Estructura y datos, datos, estructura. → Dar click en crear. → El sistema verifica y valida la información. → El sistema genera la descarga de la base de datos solicitada.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Crear BackUp Exitosamente</u> → El sistema validara la información. → El sistema descargara la base de datos.
<b>Resultados Obtenidos</b>	

<b>Camino No 1</b>	<u>Crear BackUp Exitosamente</u> → El sistema validara la información. → El sistema descargara la base de datos.
<b>Calificación Obtenida</b>	Historias Habilitadas Exitosamente

## 5.10 PRUEBAS DE ACEPTACIÓN NO. 10: CREAR UBICACIONES

Tabla 96: Pruebas de aceptación No. 10: Crear Ubicaciones

<b>Prueba de Aceptación No 10</b>	
<b>Título de la prueba</b>	Crear Ubicaciones
<b>Módulo del sistema</b>	Manejo y control
<b>Códigos de las historias</b>	HU- MC- 017
<b>Código de la prueba</b>	PRA- MC- 010
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Insertar nueva ubicación
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 0 3 si las operaciones resultan exitosas), y el Camino No1 del PRA- AU- 002 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	Insertar Nueva Ubicación → Dar click en el módulo de manejo y control → Dar click en la opción lugares y luego en la opción ubicación y en nuevo. → Deberá aparecer el formulario a diligenciar para la creación de la nueva ubicación. → Ingresar información solicitada. → Definir tipo de ubicación. → Dar click en enviar. Insertar. → El sistema generar el menaje de ubicación registrada satisfactoriamente.
<b>Resultados esperados</b>	

<b>Camino No 1</b>	Insertar Nueva Ubicación → El sistema validara la información. → Comprobara la existencia o no de la ubicación. → El sistema generara la nueva ubicación.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	Insertar Nueva Ubicación → El sistema valido la información. → El sistema generara la nueva ubicación.
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente.

## 5.11 PRUEBAS DE ACEPTACIÓN NO. 11: EDITAR UBICACIÓN

**Tabla 97: Pruebas de aceptación No. 11: Editar Ubicación**

Prueba de Aceptación No 11	
<b>Título de la prueba</b>	Editar Ubicación
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 018
<b>Código de la prueba</b>	PRA- MC- 011
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Modificar datos de ubicación previamente creado.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA-AU-003 (o el camino 2 o 3 si las operaciones no resultan exitosas).
Curso de la ejecución	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → Dar click en el módulo de Manejo y Control. → Dar click en la opción Lugares, luego en la opción ubicación, modificar. → Deberá aparecer el formulario de modificar o eliminar ubicación. → Ingresar código de ubicación. → Luego Mostrará la ubicación correspondiente al código, solicitando verificación del proceso

	<p>de eliminación.</p> <ul style="list-style-type: none"> <li>→ Cuenta con la opción editar permisos y datos de la cuenta, y cambiar contraseña.</li> <li>→ Dar click sobre el elemento a modificar.</li> <li>→ Realizar cambios.</li> <li>→ Dar click en guardar.</li> <li>→ El sistema arroja el mensaje de <i>Registro Modificado</i>.</li> <li>→</li> </ul>
<b>Camino No 2</b>	<p><u>Eliminación errónea de ubicación</u></p> <ul style="list-style-type: none"> <li>→ El sistema verificara el diligenciamiento completo de los campos.</li> <li>→ Validara la información.</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<p><u>Registro Modificado exitosamente</u></p> <ul style="list-style-type: none"> <li>→ verificación del proceso de eliminación.</li> <li>→ Dar click sobre el elemento a modificar.</li> <li>→ Realizar cambios.</li> <li>→ Dar click en guardar.</li> <li>→ El sistema editara la información solicitada.</li> </ul>
<b>Camino No 2</b>	<p><u>Eliminación errónea de ubicación</u></p> <ul style="list-style-type: none"> <li>→ El sistema verificara el diligenciamiento completo de los campos.</li> <li>→ Validara la información.</li> <li>→ Eliminará la información.</li> <li>→ Para generar nuevamente la ubicación seguir paso</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<p><u>Registro modificado Exitosamente</u></p> <ul style="list-style-type: none"> <li>→ El sistema valida la información y la acepta.</li> <li>→ El sistema elimina la ubicación.</li> </ul>
<b>Camino No 2</b>	<p><u>Eliminación Errónea de ubicación</u></p> <ul style="list-style-type: none"> <li>→ El sistema valido la información.</li> <li>→ Comprueba solicitud realizada y elimina la ubicación.</li> <li>→ Ingresar nuevamente datos de la</li> </ul>

ubicación para su creación.

## 5.12 PRUEBAS DE ACEPTACIÓN NO. 12: LISTAR UBICACIÓN

Tabla 98: Pruebas de aceptación No. 12: Listar Ubicación

<b>Prueba de Aceptación No 12</b>	
<b>Título de la prueba</b>	Listar Ubicación
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 017. HU- MC- 018, HU- MC- 019, HU- MC- 020.
<b>Código de la prueba</b>	PRA- MC-012
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Visualizar las ubicaciones ingresadas en el sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- MC- 011 (o el camino 2 o 3 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Ubicaciones</u> → Dar click en el módulo de Manejo y Control. → Dar click en la editar ubicación. → Deberá aparecer listados las ubicaciones.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Ubicaciones</u> → El sistema validara la solicitud. → Mostrará todas las ubicaciones creados en el sistema.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Ubicaciones</u> → El sistema validara la información. → El sistema listara todas las ubicaciones ingresadas.
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente.

## 5.13 PRUEBAS DE ACEPTACIÓN NO. 13: ELIMINAR UBICACIÓN

Tabla 99: Pruebas de aceptación No. 13: Eliminar Ubicación

Prueba de Aceptación No 13	
<b>Título de la prueba</b>	Eliminar Ubicación
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 017, HU- MC- 019.
<b>Código de la prueba</b>	PRA- MC-013
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Eliminar ubicaciones ingresadas en el sistema.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA- MC- 010 (o el camino 2 si las operaciones no resultan exitosas).</li> <li>☞ Contar con Role y permisos para realizar la acción de eliminación de la ubicación.</li> </ul>
Curso de la ejecución	
<b>Camino No 1</b>	<p><u>Eliminar Adecuadamente la ubicación</u></p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control.</li> <li>→ Dar click en lugares, ubicación.</li> <li>→ Dar click en la modificar.</li> <li>→ Deberá aparecer una ventana con la opción modificar o eliminar.</li> <li>→ Ingresar código de identificación de la ubicación a eliminar.</li> <li>→ Dar click en la opción eliminar la ubicación.</li> <li>→ El sistema validara la información.</li> <li>→ El sistema Mostrará una ventana con los datos de la ubicación solicitando confirmar la acción.</li> <li>→ Dar click en confirmar una vez verifíquela ubicación a eliminar.</li> <li>→ El sistema eliminara la ubicación.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de ubicación</li> </ul>

<b>Camino No 2</b>	<u>Eliminar equivocadamente la ubicación</u> → Dar click en el módulo de manejo y control. → Dar click en lugares, ubicación. → Dar click en la modificar. → Deberá aparecer una ventana con la opción modificar o eliminar. → Ingresar código de identificación de la ubicación a eliminar. → Dar click en la opción eliminar la ubicación. → El sistema validara la información. → El sistema Mostrará una ventana con los datos de la ubicación solicitando confirmar la acción. → Dar click en confirmar una vez verifíquela ubicación a eliminar. → El sistema eliminara la ubicación. → El sistema cerrara la ventana de confirmación de eliminación de ubicación → Identificar error en la eliminación → Crear nuevamente la ubicación (seguir paso)
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el usuario</u> → El sistema validara la información. → El sistema eliminara la ubicación.
<b>Camino No 2</b>	<u>Eliminar equivocadamente Ubicación</u> → El sistema elimina la ubicación. → Se identifica error. → Crear ubicación nuevamente. (Realizar paso PRA-MC-010)
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el usuario</u> → El sistema validara la información. → El sistema eliminara el usuario.
<b>Camino No 2</b>	<u>Eliminar usuario equivocado</u> → El sistema elimina el usuario. → Se identifica error. → Crear usuario nuevamente. (Realizar paso PRA-MC-010)
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente



## 5.14 PRUEBAS DE ACEPTACIÓN NO. 14: AGREGAR PADRINO

Tabla 100: Pruebas de aceptación No. 14: Agregar Padrino

Prueba de Aceptación No 14	
Título de la prueba	Agregar Padrino
Módulo del sistema	Manejo y control
Códigos de las historias	HU- MC- 022
Código de la prueba	PRA- MC- 014
Datos del caso de la prueba	
Descripción de la prueba	Insertar Padrino
Condiciones de la ejecución	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 0 3 si las operaciones no resultan exitosas), y el Camino No1 del PRA- AU- 002 (o el camino 2 si las operaciones no resultan exitosas).</li> <li>☞ Contar con Role y permisos para realizar la acción.</li> </ul>
Curso de la ejecución	
<b>Camino No 1</b>	<u>Insertar padrino</u> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control</li> <li>→ Dar click en la opción padrinos, nuevo.</li> <li>→ Deberá aparecer el formulario a diligenciar para la creación del nuevo padrino.</li> <li>→ Ingresar información solicitada.</li> <li>→ Definir código de padrino.</li> <li>→ Dar click en guardar.</li> <li>→ El sistema generar el mensaje de padrino registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<u>Debe ingresar información faltante</u> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control</li> <li>→ Dar click en la opción padrinos, nuevo.</li> <li>→ Deberá aparecer el formulario a diligenciar para la creación del nuevo padrino.</li> <li>→ Ingresar información solicitada.</li> <li>→ Definir código de padrino.</li> <li>→ Dar click en guardar.</li> <li>→ El sistema generar el mensaje de debe</li> </ul>

	ingresar información X, faltante.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	Insertar padrino → El sistema validara la información ingresada. → El sistema generar el mensaje de padrino registrado satisfactoriamente.
<b>Camino No 2</b>	Debe ingresar información faltante → Al validar la información el sistema evidencia falta de diligenciamiento de la misma. → El sistema generar el mensaje de debe ingresar información X, faltante.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	Insertar padrino → El sistema valido la información ingresada. → El sistema genera el mensaje de padrino registrado satisfactoriamente.
<b>Camino No 2</b>	Debe ingresar información faltante → valida la información y evidencia falta de diligenciamiento de la misma. → El sistema generar el mensaje de debe ingresar información X, faltante. → Se ingresa información, el sistema genera el mensaje de padrino registrado adecuadamente.
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente

## 5.15 PRUEBAS DE ACEPTACIÓN NO. 15: EDITAR PADRINO

Tabla 101: Pruebas de aceptación No. 15: Editar padrino

<b>Prueba de Aceptación No 15</b>	
<b>Título de la prueba</b>	Editar Padrino
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 022, HU- MC- 023
<b>Código de la prueba</b>	PRA- AU- 015
<b>Datos del caso de la prueba</b>	

<b>Descripción de la prueba</b>	Modificar datos de padrino previamente creado.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA-MC-014 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → Dar click en el módulo de Manejo y Control. → Dar click en la opción padrino, luego en la opción modificar. → Deberá aparecer el formulario de modificar o eliminar padrino. → Ingresar código de padrino. → Luego Mostrará el padrino correspondiente al código. → Muestra la opción editar. → Dar click en editar. → Modificar información del padrino. → Guardar cambios realizados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro modificado Exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente

## 5.16 PRUEBAS DE ACEPTACIÓN NO. 16: LISTAR PADRINO

Tabla 102: Pruebas de aceptación No. 16: Listar padrino

Pruebas de aceptación No. 16: Listar padrino	
<b>Título de la prueba</b>	Listar Padrino
<b>Módulo del sistema</b>	Manejo y Control

<b>Códigos de las historias</b>	HU- MC- 022, HU- MC- 023, HU- MC- 024, HU- MC- 025.
<b>Código de la prueba</b>	PRA- MC- 016
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Visualizar los padrinos ingresados en el sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- MC-014 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente padrino</u> → Dar click en el módulo de Manejo y Control. → Dar click en padrino, listar. → Deberá aparecer listados los padrinos.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Padrino</u> → El sistema validara la solicitud. → Mostrará todos los padrinos creados en el sistema.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente Padrino</u> → El sistema validara la información. → El sistema listo todos los padrinos ingresados.
<b>Calificación Obtenida</b>	Historias Habilitadas Adecuadamente

## 5.17 PRUEBAS DE ACEPTACIÓN NO. 17: ELIMINAR PADRINO

Tabla 103: Pruebas de aceptación No. 17: Eliminar Padrino

Prueba de Aceptación No 17	
<b>Título de la prueba</b>	Eliminar Padrino
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 022, HU- MC- 024
<b>Código de la prueba</b>	PRA- MC- 017
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Eliminar Padrino ingresados en el

	sistema.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA- MC- 014 (o el camino 2 si las operaciones no resultan exitosas).</li> <li>☞ Contar con los permisos para realizar la acción.</li> </ul>
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el padrino</u> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control.</li> <li>→ Dar click en padrino.</li> <li>→ Dar click en la modificar.</li> <li>→ Deberá aparecer una ventana con la opción modificar o eliminar.</li> <li>→ Ingresar código del padrino a eliminar.</li> <li>→ Ingresar código.</li> <li>→ Dar click en la opción seleccionar.</li> <li>→ El sistema genera el mensaje de Elija una acción entre modificar y eliminar.</li> <li>→ Dar click en la opción eliminar padrino.</li> <li>→ El sistema validara la información.</li> <li>→ El sistema Mostrará una ventana con los datos de la ubicación solicitando confirmar la acción.</li> <li>→ Dar click en confirmar.</li> <li>→ El sistema eliminara el padrino.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de padrino.</li> </ul>
<b>Camino No 1</b>	<u>Eliminar equivocadamente padrino</u> <ul style="list-style-type: none"> <li>→ El sistema eliminara el padrino.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación del padrino.</li> <li>→ Crear nuevamente el padrino (seguir paso PRA-MC-014)</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el Padrino</u> <ul style="list-style-type: none"> <li>→ El sistema validara la información.</li> <li>→ El sistema eliminara el padrino.</li> </ul>

<b>Camino No 2</b>	<u>Eliminar equivocadamente padrino</u> → El sistema eliminara el padrino. → Crear padrino nuevamente. (Realizar paso PRA-MC-014)
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el apadrinamiento</u> → El sistema validara la información. → El sistema eliminara el apadrinamiento.
<b>Camino No 2</b>	<u>Eliminar apadrinamiento equivocado</u> → El sistema eliminara el apadrinamiento. → Se identifica error. → Crear apadrinamiento nuevamente. (Realizar paso PRA-MC-014)
<b>Calificación obtenida</b>	Historias Habilitadas Satisfactoriamente

## 5.18 PRUEBAS DE ACEPTACIÓN NO. 18: AGREGAR INFORMACIÓN NIÑO

**Tabla 104: Pruebas de aceptación No. 18: Agregar Información Niño**

<b>Prueba de Aceptación No 18</b>	
<b>Título de la prueba</b>	Agregar Información Niño
<b>Módulo del sistema</b>	Manejo y control
<b>Códigos de las historias</b>	HU- MC- 027
<b>Código de la prueba</b>	PRA- MC- 018
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Insertar niño.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 o 3 si las operaciones resultan exitosas), y el Camino No1 del PRA- AU- 002 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	Crear nuevo niño → Dar click en el módulo de manejo y control → Dar click en la opción ahijado, nuevo. → Deberá aparecer el formulario a

	<p>diligenciar para la creación del nuevo ahijado.</p> <ul style="list-style-type: none"> <li>→ Ingresar información solicitada.</li> <li>→ Completar la información.</li> <li>→ Dar click en guardar.</li> <li>→ El sistema generar el mensaje de ahijado registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<p>Debe ingresar información faltante</p> <ul style="list-style-type: none"> <li>→ Dar click en guardar.</li> <li>→ El sistema generar el mensaje de debe ingresar información X, faltante.</li> <li>→ El sistema no cierra el formulario.</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<p>Crear nuevo niño</p> <ul style="list-style-type: none"> <li>→ El sistema valide la información.</li> <li>→ El sistema generar el mensaje de ahijado registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<p>Debe ingresar información faltante</p> <ul style="list-style-type: none"> <li>→ El sistema generar el mensaje de debe ingresar información X, faltante.</li> <li>→ El sistema no cierra el formulario.</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<p>Crear nuevo niño</p> <ul style="list-style-type: none"> <li>→ El sistema valido la información.</li> <li>→ El sistema genero el mensaje de ahijado registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<p>Debe ingresar información faltante</p> <ul style="list-style-type: none"> <li>→ El sistema genero el mensaje de debe ingresar información X, faltante.</li> <li>→ Ingresa información faltante.</li> <li>→ El sistema cerró el formulario satisfactoriamente.</li> </ul>
<b>Calificación obtenida</b>	Historias Habilitadas satisfactoriamente

## 5.19 PRUEBAS DE ACEPTACIÓN NO. 19: EDITAR INFORMACIÓN NIÑO

Tabla 105: Pruebas de aceptación No. 19: Editar Información Niño

Prueba de Aceptación No 19	
<b>Título de la prueba</b>	Editar Información Niño
<b>Módulo del sistema</b>	Manejo y Control

<b>Códigos de las historias</b>	HU- MC- 027, HU- MC- 028
<b>Código de la prueba</b>	PRA- MC-019
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Modificar datos de ahijado previamente creado.
<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA-MC-018 (o el camino 2 si las operaciones resultan exitosas).</li> <li>☞ Contar con permiso para realizar este proceso.</li> </ul>
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → Dar click en el módulo de Manejo y Control. → Dar click en la opción ahijado, luego en la opción modificar. → Deberá aparecer el formulario de modificar o eliminar ahijado. → Ingresar código de ahijado. → Luego Mostrará el padrino correspondiente al código. → Muestra la opción editar. → Dar click en editar. → Modificar información del padrino. → Guardar cambios realizados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro modificado Exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Calificación obtenida</b>	Historias Habilitadas satisfactoriamente



## 5.20 PRUEBAS DE ACEPTACIÓN NO. 20: ELIMINAR AHIJADO

Tabla 106: Pruebas de aceptación No. 20: Eliminar ahijado

Prueba de Aceptación No 20	
Título de la prueba	Eliminar ahijado
Módulo del sistema	Manejo y Control
Códigos de las historias	HU- MC- 027,HU- MC- 028, HU- MC- 029
Código de la prueba	PRA- MC-020
Datos del caso de la prueba	
Descripción de la prueba	Eliminar Padrino ingresados en el sistema.
Condiciones de la ejecución	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA- MC-018 (o el camino 2 si las operaciones resultan exitosas).</li> <li>☞ Contar con permisos para realizar dicha acción.</li> </ul>
Curso de la ejecución	
Camino No 1	<p><u>Eliminar Adecuadamente el ahijado</u></p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control.</li> <li>→ Dar click en ahijado.</li> <li>→ Dar click en listar.</li> <li>→ Se listan todos los niños ingresados.</li> <li>→ Aparece la opción listar al lado de cada niño.</li> <li>→ Dar click en eliminar.</li> <li>→ El sistema Mostrará una ventana con los datos del ahijado.</li> <li>→ solicitando confirmar la acción.</li> <li>→ Dar click en confirmar, una vez confirme verifique el ahijado a eliminar.</li> <li>→ El sistema eliminara al ahijado.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de ahijado.</li> </ul>
Camino No 1	<p><u>Eliminar ahijado equivocado</u></p> <ul style="list-style-type: none"> <li>→ El sistema eliminara al ahijado.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de ahijado.</li> <li>→ Identificar error en la eliminación</li> </ul>

	→ Crear nuevamente el ahijado (seguir paso PRA-MC-018)
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el ahijado</u> → El sistema validara la información. → El sistema eliminara al ahijado.
<b>Camino No 2</b>	<u>Eliminar ahijado equivocado</u> → El sistema eliminara el ahijado. → Se identifica error. → Crear ahijado nuevamente. (Realizar paso PRA-MC-018)
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el ahijado</u> → El sistema validara la información. → El sistema eliminara al ahijado.
<b>Camino No 2</b>	<u>Eliminar usuario equivocado</u> → El sistema elimina al ahijado → Se identifica error. → Crear ahijado nuevamente. (Realizar paso PRA-MC-018)
<b>Calificación Obtenida</b>	Historias Habilitadas Exitosamente

## 5.21 PRUEBAS DE ACEPTACIÓN NO. 21: LISTAR NIÑO

Tabla 107: Pruebas de aceptación No. 21: Listar niño

Prueba de Aceptación No 21	
<b>Título de la prueba</b>	Listar ahijado
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 027
<b>Código de la prueba</b>	PRA- MC- 021
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Visualizar los ahijados ingresados en el sistema.

<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- MC-018 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente ahijado</u> → Dar click en el módulo de Manejo y Control. → Dar click en ahijado, listar. → Deberá aparecer listados los ahijados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente ahijado</u> → El sistema validara la solicitud. → Mostrará todos los ahijados creados en el sistema.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente ahijado</u> → El sistema valido la solicitud. → Muestra todos los ahijados creados en el sistema.
<b>Calificación obtenida</b>	Historias habilitadas satisfactoriamente

## 5.22 PRUEBAS DE ACEPTACIÓN NO. 22: CREAR APADRINAMIENTO

Tabla 108: Pruebas de aceptación No. 22: Crear Apadrinamiento

Prueba de Aceptación No 22	
<b>Título de la prueba</b>	Crear Apadrinamiento
<b>Módulo del sistema</b>	Manejo y control
<b>Códigos de las historias</b>	HU- MC- 032
<b>Código de la prueba</b>	PRA- MC- 022
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Insertar proceso de apadrinamiento.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 o 3 si las operaciones no resultan exitosas), y el Camino No1 del PRA- AU- 002 (o el

	camino 2 si las operaciones no resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<p>Crear nuevo apadrinamiento</p> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control</li> <li>→ Dar click en la opción apadrinamiento, apadrinar, nuevo.</li> <li>→ Deberá aparecer el formulario a diligenciar para la creación del nuevo apadrinamiento.</li> <li>→ Ingresar información solicitada.</li> <li>→ Completar la información.</li> <li>→ Dar click en guardar.</li> <li>→ El sistema generar el mensaje de apadrinamiento registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<p>Debe ingresar información faltante</p> <ul style="list-style-type: none"> <li>→ Dar click en guardar.</li> <li>→ El sistema generar el mensaje de debe ingresar información X, faltante.</li> <li>→ El sistema no cierra el formulario.</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<p>Crear nuevo apadrinamiento</p> <ul style="list-style-type: none"> <li>→ El sistema valide la información.</li> <li>→ El sistema generar el mensaje de apadrinamiento registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<p>Debe ingresar información faltante</p> <ul style="list-style-type: none"> <li>→ El sistema generar el mensaje de debe ingresar información X, faltante.</li> <li>→ El sistema no cierra el formulario.</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<p>Crear nuevo apadrinamiento</p> <ul style="list-style-type: none"> <li>→ El sistema valido la información.</li> <li>→ El sistema genero el mensaje de apadrinamiento, registrado satisfactoriamente.</li> </ul>
<b>Camino No 2</b>	<p>Debe ingresar información faltante</p> <ul style="list-style-type: none"> <li>→ El sistema genero el mensaje de debe ingresar información X, faltante.</li> <li>→ Ingresa información completa.</li> </ul>

	→ El sistema cerró el formulario.
<b>Calificación obtenida</b>	Historias habilitadas satisfactoriamente

## 5.23 PRUEBAS DE ACEPTACIÓN NO. 23: EDITAR APADRINAMIENTO

Tabla 109: Pruebas de aceptación No. 23: Editar Apadrinamiento

<b>Prueba de Aceptación No 23</b>	
<b>Título de la prueba</b>	Editar Apadrinamiento
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 032, HU- MC- 033
<b>Código de la prueba</b>	PRA- MC-023
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Modificar datos de apadrinamiento previamente creado.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA-MC-022 (o el camino 2 o 3 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → Dar click en el módulo de Manejo y Control. → Dar click en la opción ahijado, luego en la opción modificar. → Deberá aparecer el formulario de modificar o eliminar ahijado. → Ingresar código de ahijado. → Luego Mostrará el padrino correspondiente al código. → Muestra la opción editar. → Dar click en editar. → Modificar información del padrino. → Guardar cambios realizados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → El sistema valido la información. → Incorpora los cambios.

	→ Genera informe de la modificación realizada.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro modificado Exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Calificación obtenida</b>	Historias habilitadas satisfactoriamente

## 5.24 PRUEBAS DE ACEPTACIÓN NO. 24: ELIMINAR APADRINAMIENTO

Tabla 110: Pruebas de aceptación No. 24: Eliminar apadrinamiento

Prueba de Aceptación No 24	
<b>Título de la prueba</b>	Eliminar apadrinamiento
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 032,HU- MC- 033, HU- MC- 034
<b>Código de la prueba</b>	PRA- MC-024
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Eliminar procesos de apadrinamiento ingresados en el sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- MC- 022 (o el camino 2 si las operaciones resultan exitosas). ☞ Contar con permiso para realizar esta acción.
Curso de la ejecución	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el proceso de apadrinamiento</u> → Dar click en el módulo de manejo y control. → Dar click en apadrinamiento → Dar click en listar. → Se listan todos los procesos ingresados. → Aparece la opción listar al lado de cada niño. → Dar click en eliminar. → El sistema Mostrará una ventana con los datos de la ubicación

	<p>solicitando confirmar la acción.</p> <ul style="list-style-type: none"> <li>→ Dar click en confirmar, una vez confirme verifique el apadrinamiento a eliminar.</li> <li>→ El sistema eliminara el apadrinamiento.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de apadrinamiento.</li> </ul>
<b>Camino No 1</b>	<p><u>Eliminar Apadrinamiento equivocado</u></p> <ul style="list-style-type: none"> <li>→ El sistema eliminara al apadrinamiento.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de apadrinamiento.</li> <li>→ Crear nuevamente el proceso de apadrinamiento (seguir paso PRA-MC-022)</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<p><u>Eliminar Adecuadamente el apadrinamiento</u></p> <ul style="list-style-type: none"> <li>→ El sistema validara la información.</li> <li>→ El sistema eliminara el apadrinamiento.</li> </ul>
<b>Camino No 2</b>	<p><u>Eliminar apadrinamiento equivocado</u></p> <ul style="list-style-type: none"> <li>→ El sistema eliminara el apadrinamiento.</li> <li>→ Se identifica error.</li> <li>→ Crear apadrinamiento nuevamente. (Realizar paso PRA-MC-022)</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<p><u>Eliminar Adecuadamente el apadrinamiento</u></p> <ul style="list-style-type: none"> <li>→ El sistema validara la información.</li> <li>→ El sistema eliminara el apadrinamiento.</li> </ul>
<b>Camino No 2</b>	<p><u>Eliminar apadrinamiento equivocado</u></p> <ul style="list-style-type: none"> <li>→ El sistema elimina al apadrinamiento</li> <li>→ Se identifica error.</li> <li>→ Crear apadrinamiento nuevamente. (Realizar paso PRA-MC-022)</li> </ul>
<b>Calificación obtenida</b>	Historias habilitadas satisfactoriamente

## 5.25 PRUEBAS DE ACEPTACIÓN NO. 25: LISTAR APADRINAMIENTO

Tabla 111: Pruebas de aceptación No. 25: Listar apadrinamiento

<b>Prueba de Aceptación No 25</b>	
<b>Título de la prueba</b>	Listar ahijado
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 032, HU- MC- 033, HU- MC- 034, HU- MC- 035.
<b>Código de la prueba</b>	PRA- MC-025
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Visualizar los ahijados ingresados en el sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- MC- 022 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente apadrinamiento</u> → Dar click en el módulo de Manejo y Control. → Dar click en apadrinamiento, listar. → Deberá aparecer listados los procesos de apadrinamiento.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente apadrinamiento</u> → El sistema validara la solicitud. → Mostrará todos los apadrinamientos creados en el sistema.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Listar adecuadamente apadrinamiento</u> → El sistema valido la solicitud. → Muestra todo el apadrinamiento creados en el sistema.
<b>Calificación Obtenida</b>	Historias Habilitadas Satisfactoriamente.



## 5.26 PRUEBAS DE ACEPTACIÓN NO. 26: AGREGAR DIBUJO

Tabla 111: Pruebas de aceptación No. 26: Agregar Dibujo

<b>Prueba de Aceptación No 26</b>	
<b>Título de la prueba</b>	Agregar Dibujo
<b>Módulo del sistema</b>	Manejo y control
<b>Códigos de las historias</b>	HU- MC- 037
<b>Código de la prueba</b>	PRA- MC- 026
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Insertar proceso de dibujo a campaña.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 0 3 si las operaciones resultan exitosas), y el Camino No1 del PRA- AU- 002 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Crear Dibujo</u> → Dar click en el módulo de manejo y control → Dar click en la opción campaña dibujo, campaña dibujo. → Aparecerá a lista de los dibujos ingresados por niño. → En la parte superior izquierda se visualiza el botón agregar dibujo. → Deberá aparecer el formulario a diligenciar para la creación del nuevo dibujo. → Ingresar información solicitada. → Completar la información. → Dar click en guardar. → El sistema generar el mensaje de nuevo dibujo ingresado adecuadamente.
<b>Camino No 2</b>	<u>Debe ingresar información faltante</u> → Dar click en guardar. → El sistema generar el mensaje de debe ingresar información X, faltante. → El sistema no cierra el formulario.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Crear Dibujo</u> → El sistema valide la información.

	→ El sistema generar el mensaje de dibujo registrado satisfactoriamente.
<b>Camino No 2</b>	<u>Debe ingresar información faltante</u> → El sistema generar el mensaje de debe ingresar información X, faltante. → El sistema cierra el formulario.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Crear Dibujo</u> → El sistema valide la información. → El sistema generar el mensaje de dibujo registrado satisfactoriamente.
<b>Camino No 2</b>	<u>Debe ingresar información faltante</u> → El sistema generar el mensaje de debe ingresar información X, faltante. → El sistema no cierra el formulario.
<b>Calificación obtenida</b>	Historias Habilitadas satisfactoriamente

## 5.27 PRUEBAS DE ACEPTACIÓN NO. 27: EDITAR DIBUJO

Tabla 112: Pruebas de aceptación No. 27: Editar Dibujo

Prueba de Aceptación No 27	
<b>Título de la prueba</b>	Editar Dibujo
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HHU- MC- 037 HU- MC- 038
<b>Código de la prueba</b>	PRA- MC-027
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Modificar datos de campaña de dibujo previamente creado.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA-MC-026 (o el camino 2 o 3 si las operaciones resultan exitosas).
Curso de la ejecución	

<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → Dar click en el módulo de Manejo y Control. → Dar click en la opción campaña de dibujo, campaña de dibujo. → ahijado, luego en la opción modificar. → Deberá aparecer el formulario de modificar o eliminar dibujo. → Muestra la opción editar o modificar. → Dar click en dibujo → Modificar información del dibujo → Guardar cambios realizados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Registro Modificado exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Registro modificado Exitosamente</u> → El sistema valido la información. → Incorpora los cambios. → Genera informe de la modificación realizada.
<b>Calificación Obtenida</b>	Habilitar Historias Satisfactoriamente.

## 5.28 PRUEBAS DE ACEPTACIÓN NO. 28: ELIMINAR DIBUJO

Tabla 113: Pruebas de aceptación No. 28: Eliminar Dibujo

Prueba de Aceptación No 28	
<b>Título de la prueba</b>	Eliminar Dibujo
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 037, HU- MC- 039.
<b>Código de la prueba</b>	PRA- MC-028
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Eliminar dibujado ingresado en el sistema.

<b>Condiciones de la ejecución</b>	<ul style="list-style-type: none"> <li>☞ Haber seguido el Camino No1 del PRA- MC- 026 (o el camino 2 si las operaciones no resultan exitosas).</li> <li>☞ Contar con los permisos para realizar esta acción.</li> </ul>
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el dibujo</u> <ul style="list-style-type: none"> <li>→ Dar click en el módulo de manejo y control.</li> <li>→ Dar click en campaña de dibujo.</li> <li>→ Dar click en listar.</li> <li>→ Se listan todos los dibujos ingresados.</li> <li>→ Aparece la opción eliminar al lado de cada niño.</li> <li>→ Dar click en eliminar.</li> <li>→ El sistema Mostrará una ventana con los datos del dibujo solicitando confirmar la acción.</li> <li>→ Dar click en confirmar, una vez confirme identificar dibujo a eliminar.</li> <li>→ El sistema eliminara el dibujo.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de Dibujo.</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el dibujo</u> <ul style="list-style-type: none"> <li>→ El sistema valido la información.</li> <li>→ El sistema cerrara la ventana de confirmación de eliminación de Dibujo.</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Eliminar Adecuadamente el dibujo</u> <ul style="list-style-type: none"> <li>→ El sistema valido la información.</li> <li>→ El sistema cerró la ventana de confirmación de eliminación de Dibujo.</li> </ul>
<b>Calificación obtenida</b>	Historia Habilidades Satisfactoriamente.

## 5.29 PRUEBAS DE ACEPTACIÓN NO. 29: LISTAR DIBUJO

Tabla 114: Pruebas de aceptación No. 29: Listar Dibujo

<b>Prueba de Aceptación No 29</b>	
<b>Título de la prueba</b>	Listar Dibujo
<b>Módulo del sistema</b>	Manejo y Control
<b>Códigos de las historias</b>	HU- MC- 037, HU- MC- 040
<b>Código de la prueba</b>	PRA- MC-029
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Mostrar dibujos ingresado en el sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- MC-026 (o el camino 2 si las operaciones resultan exitosas).
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Listar dibujo exitosamente</u> → Dar click en el módulo de manejo y control. → Dar click en campaña de dibujo. → Dar click en listar. → Se listan todos los dibujos ingresados.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Listar dibujo exitosamente</u> → El sistema validara la información. → El sistema listara los dibujos ingresados.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Listar dibujo exitosamente</u> → El sistema valido la información. → El sistema listo los dibujos ingresados.
<b>Calificación obtenida</b>	Historia Habilitadas Satisfactoriamente.

### 5.30 PRUEBAS DE ACEPTACIÓN NO. 030: EXPORTAR DATOS A ESPAÑA

Tabla 115: Pruebas de aceptación No. 030: Exportar Datos a España

Prueba de Aceptación No 30	
<b>Título de la prueba</b>	Exportar Datos a España
<b>Módulo del sistema</b>	Exportar
<b>Códigos de las historias</b>	HU- EX 043
<b>Código de la prueba</b>	PRA- EX - 030
Datos del caso de la prueba	
<b>Descripción de la prueba</b>	Mostrar información ingresada al sistema de información.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 o 3 si las operaciones resultan exitosas). Haber seguido el Camino No1 del PRA- AU- 002 (o el camino 2 o 3 si las operaciones resultan exitosas).y todos los PRA del sistema de información.
Curso de la ejecución	
<b>Camino No 1</b>	<u>Generar Base de Datos bajas exitosamente</u> → Dar click en el módulo de exportar → Dar click en base a España. → Seleccionar la opción deseada: sin bajas, con bajas o completa según necesidad y/o preferencia. → El sistema generara la base de datos online.
<b>Camino No 2</b>	<u>Generar Base de Datos sin bajas exitosamente</u> → Dar click en el módulo de exportar → Dar click en base a España. → Dar click en bajas El sistema generara la base de datos online.
<b>Camino No 3</b>	<u>Generar Base de Datos completas exitosamente</u> → Dar click en el módulo de exportar → Dar click en base a España. → Dar click en completas → El sistema generara la base de

	datos online.
<b>Camino No 4</b>	<u>Problemas la generar base de datos</u> → Dar click en el módulo de exportar → Dar click en sin bajas, bajas o completas → El sistema no genera la base de datos, genera error al exportar → Iniciar sesión nuevamente y generar proceso de exportar.
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Generar Base de Datos Sin bajas exitosamente</u> → El sistema validara la información. → El sistema generara base de datos online.
<b>Camino No 2</b>	<u>Generar Base de Datos bajas exitosamente</u> → El sistema validara la información. → El sistema generara base de datos online.
<b>Camino No 3</b>	Generar Base de Datos completas exitosamente → El sistema validara la información. → El sistema generara base de datos online.
<b>Camino No 4</b>	<u>Problemas la generar base de datos</u> → El sistema no validara la información. → El sistema generara error. → Iniciar sesión nuevamente.
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Generar Base de Datos Sin bajas exitosamente</u> → El sistema valido la información. → El sistema genero base de datos online.
<b>Camino No 2</b>	<u>Generar Base de Datos bajas exitosamente</u> → El sistema valido la información. → El sistema genero base de datos online.

<b>Camino No 3</b>	Generar Base de Datos completas exitosamente → El sistema valido la información. → El sistema genero base de datos online.
<b>Camino No 4</b>	<u>Problemas la generar base de datos</u> → El sistema no valido la información. → El sistema genero error. → Se Inició sesión nuevamente. → El sistema genera la base de datos.
<b>Calificación obtenida</b>	Historias habilitadas exitosamente.

### 5.31 PRUEBAS DE ACEPTACIÓN NO. 031: REPORTAR BITÁCORA DE USO DE SIGLHA

**Tabla 116: Pruebas de aceptación No. 031: Reportar Bitácora de Uso de SIGLHA**

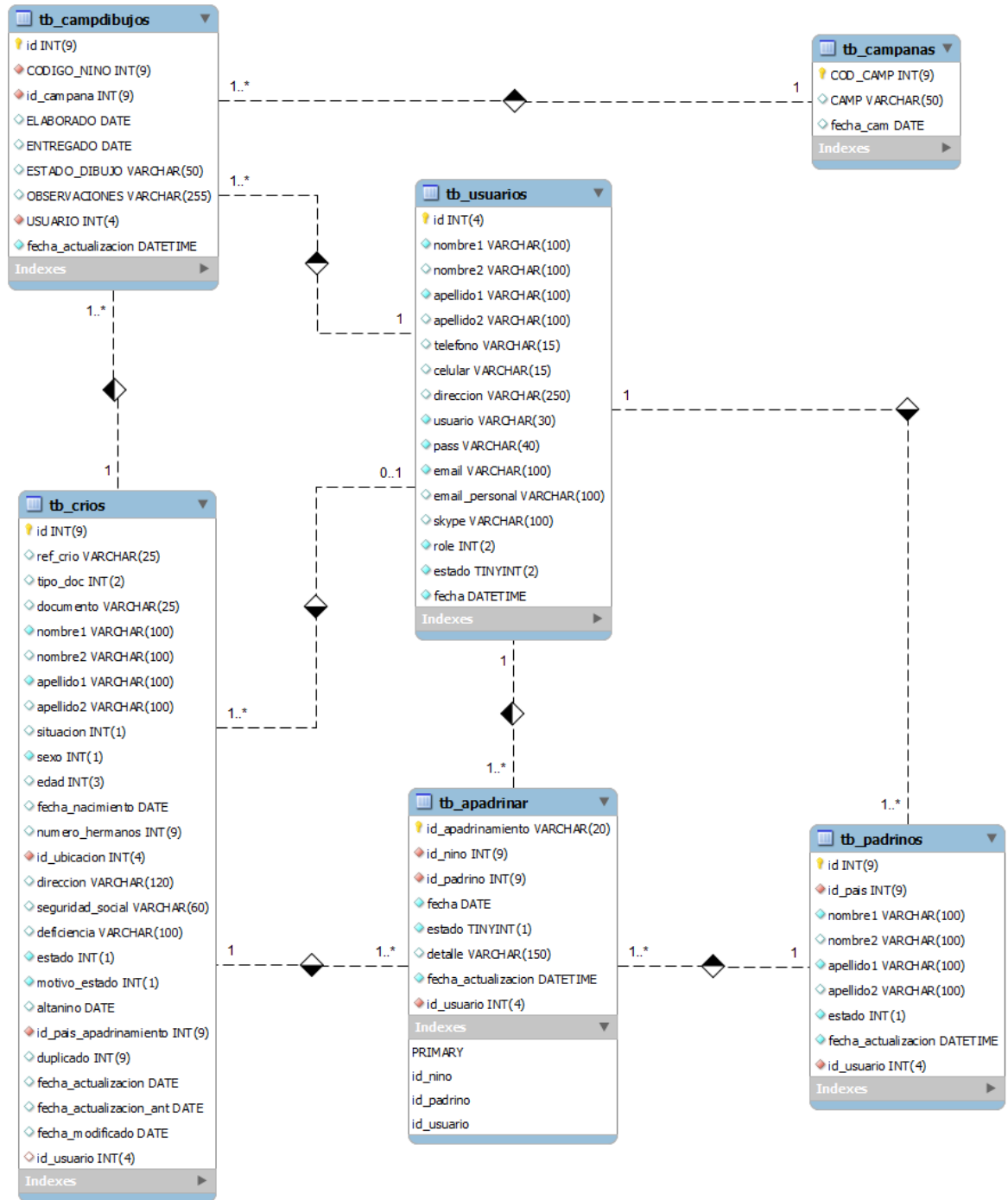
<b>Prueba de Aceptación No 31</b>	
<b>Título de la prueba</b>	Reportar Bitácora de Uso de SIGLHA
<b>Módulo del sistema</b>	Reportes/Informes
<b>Códigos de las historias</b>	HU- RI- 044
<b>Código de la prueba</b>	PRA- RI- 031
<b>Datos del caso de la prueba</b>	
<b>Descripción de la prueba</b>	Conocer ingresos de los usuarios del sistema.
<b>Condiciones de la ejecución</b>	☞ Haber seguido el Camino No1 del PRA- AU- 001 (o el camino 2 o 3 si las operaciones resultan exitosas). Haber seguido el Camino No1 del PRA- AU- 002 (o el camino 2 o 3 si las operaciones resultan exitosas).y todos los PRA del sistema de información.
<b>Curso de la ejecución</b>	
<b>Camino No 1</b>	<u>Reporte de Actividad Exitoso</u> → Dar click en el Reporte/informes → Dar click en actividad → El sistema Mostrará el formulario a diligenciar.

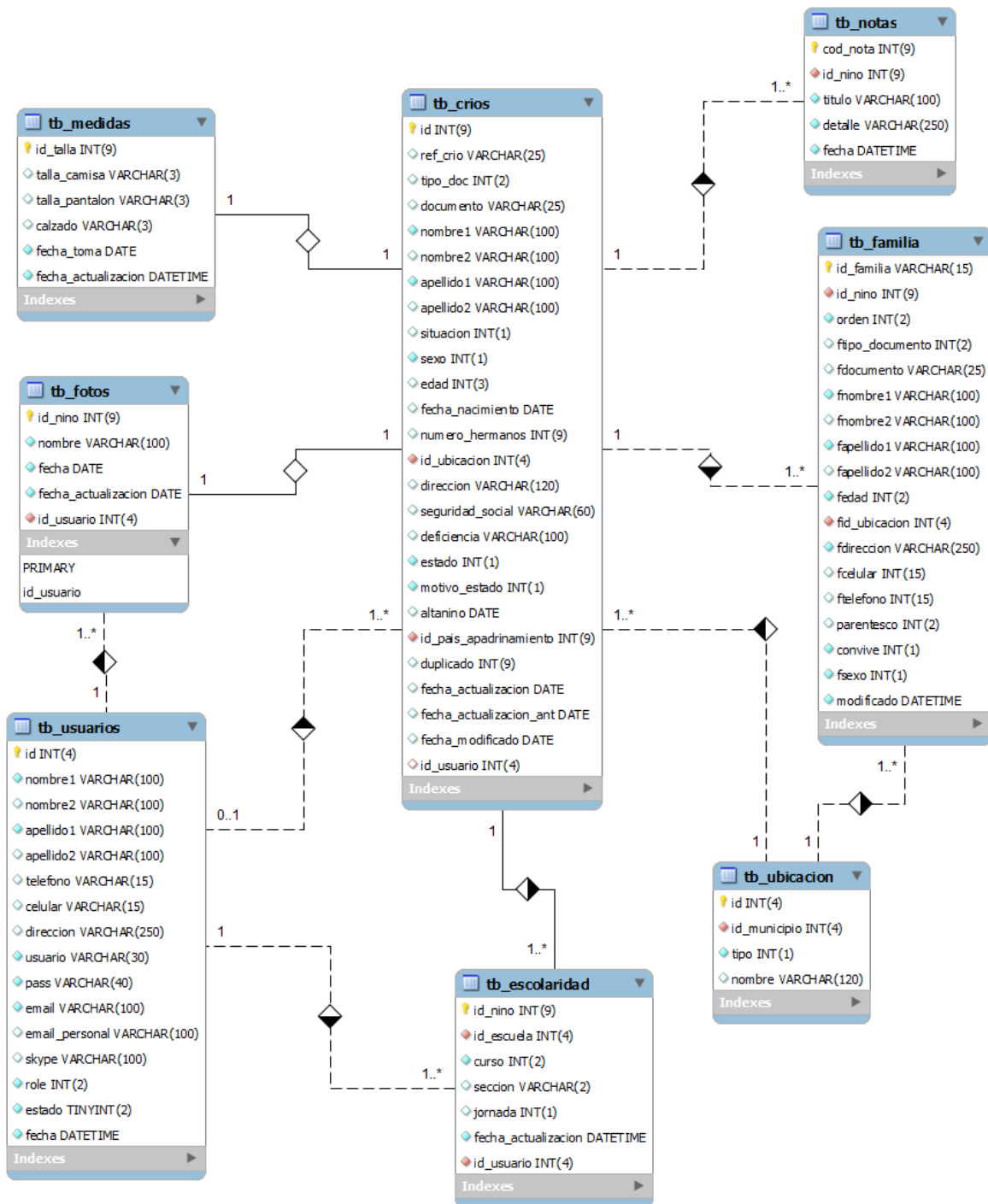


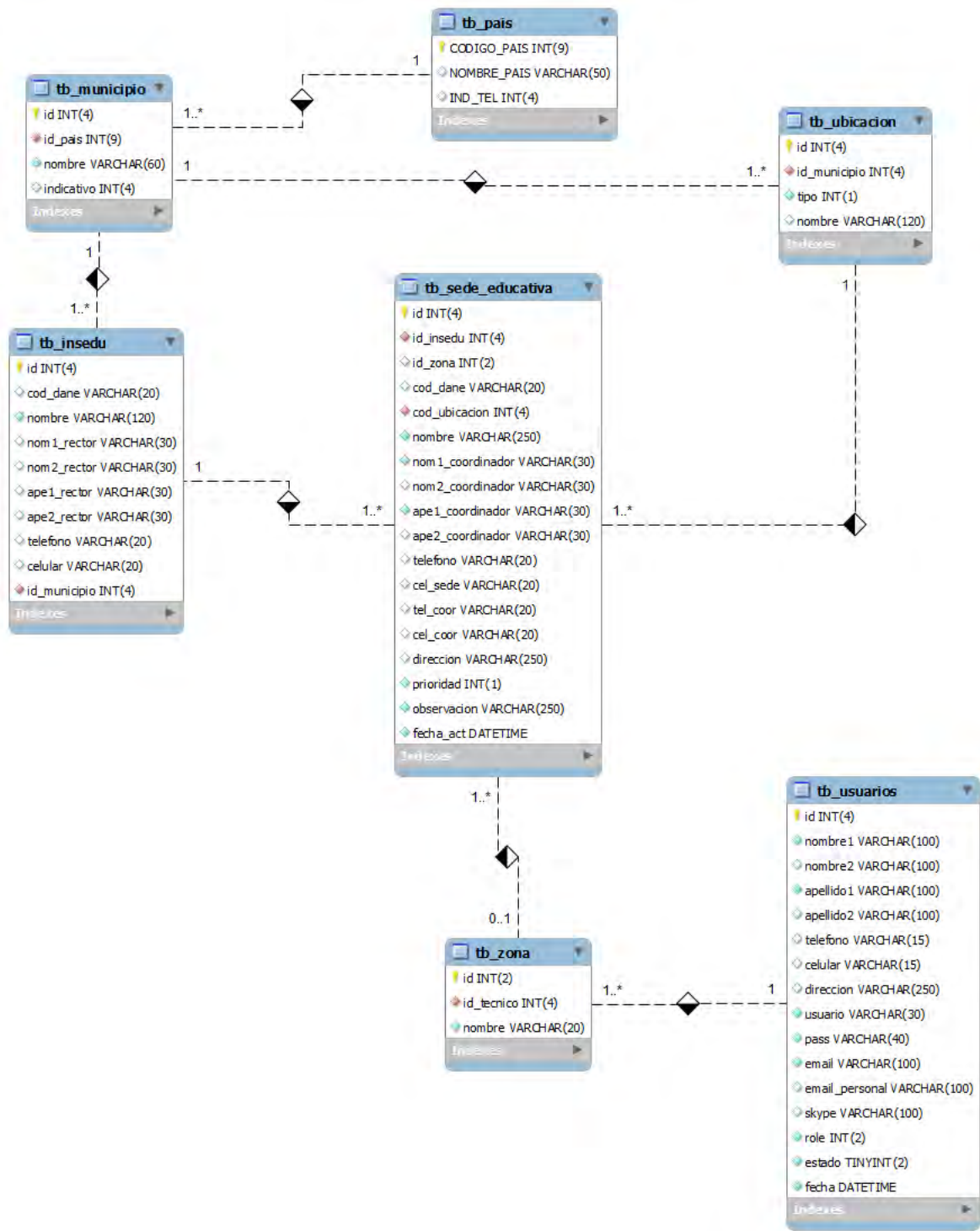
	<ul style="list-style-type: none"> <li>→ Diligencie el formulario, con el código del usuario y mes a reportar actividades.</li> <li>→ Dar click en mostrar actividades.</li> </ul>
<b>Camino No 2</b>	<u>Reporte en Blanco</u> <ul style="list-style-type: none"> <li>→ Error al digitar código de usuario.</li> <li>→ Verifique nuevamente el código, e intente nuevamente.</li> <li>→ Verifique mes a reportar actividades</li> <li>→ Comuníquese con el administrador.</li> </ul>
<b>Resultados esperados</b>	
<b>Camino No 1</b>	<u>Reporte de Actividad Exitoso</u> <ul style="list-style-type: none"> <li>→ El sistema compara la información ingresada con la almacenada.</li> <li>→ Sistema valida correctamente la información.</li> <li>→ El sistema muestra actividades ejecutada por los usuarios.</li> </ul>
<b>Camino No 2</b>	<u>Reporte en Blanco</u> <ul style="list-style-type: none"> <li>→ El sistema compara la información ingresada con la almacenada.</li> <li>→ Sistema no valida correctamente la información.</li> <li>→ El sistema genera un reporte en blanco.</li> </ul>
<b>Resultados Obtenidos</b>	
<b>Camino No 1</b>	<u>Reporte de Actividad Exitoso</u> <ul style="list-style-type: none"> <li>→ El Sistema valido correctamente la información.</li> <li>→ El sistema mostro actividades ejecutada por los usuarios.</li> </ul>
<b>Camino No 2</b>	<u>Reporte en Blanco</u> <ul style="list-style-type: none"> <li>→ El sistema comparo la información ingresada con la almacenada.</li> <li>→ El sistema genera un reporte en blanco.</li> <li>→ Verificar código.</li> <li>→ Intentar de nuevo.</li> <li>→ Lista informe.</li> </ul>
<b>Calificación obtenida</b>	Historias habilitadas satisfactoriamente.

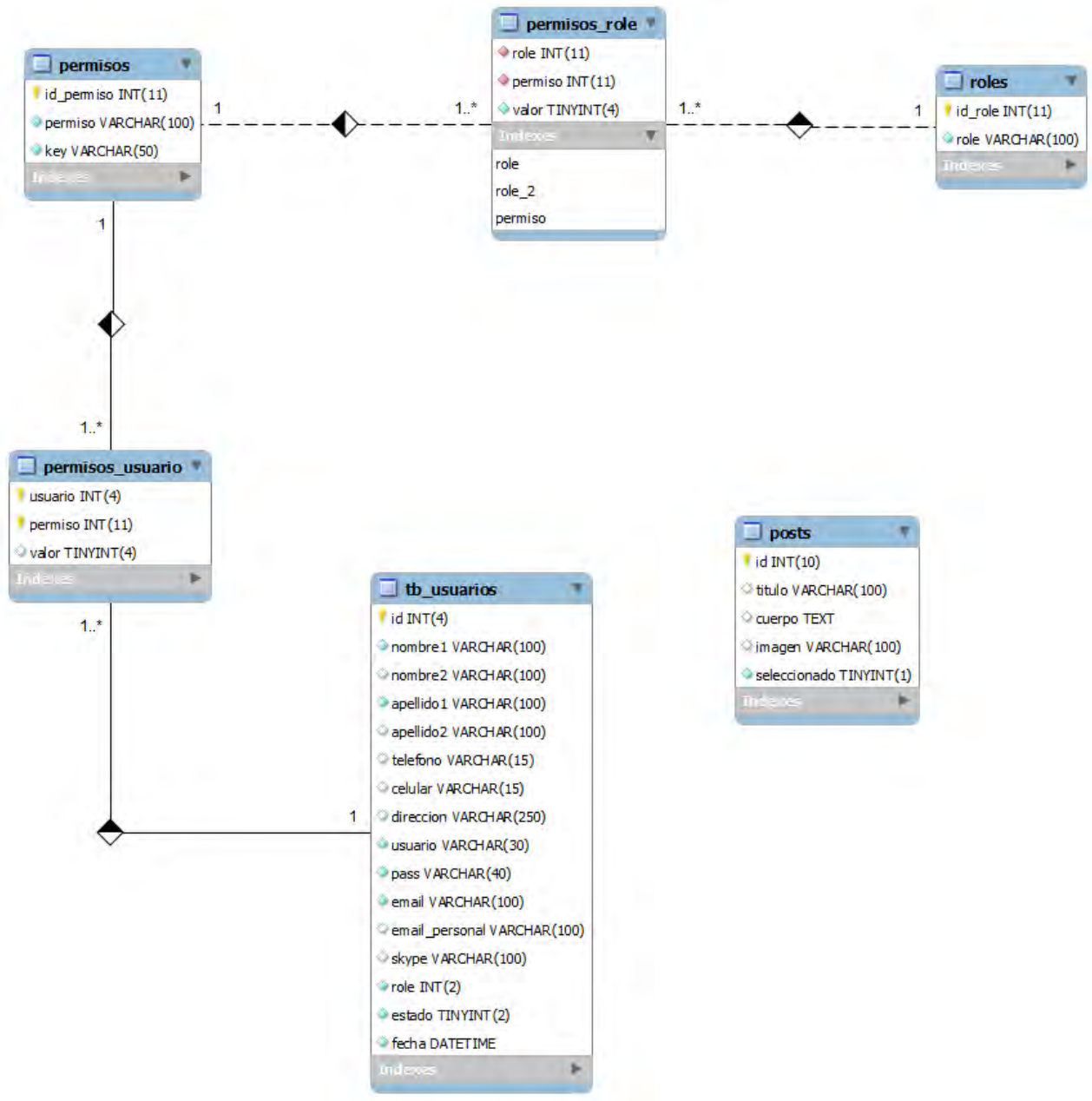
## 6. DISEÑO DE DATOS

Figura 5: Entidad- Relación









## 7. MODELO DE CLASES

Figura 6: Modelo de Clases.



## 8. DIAGRAMA DE PAQUETES

Figura 7: Diagramas de paquetes - SIGLHA

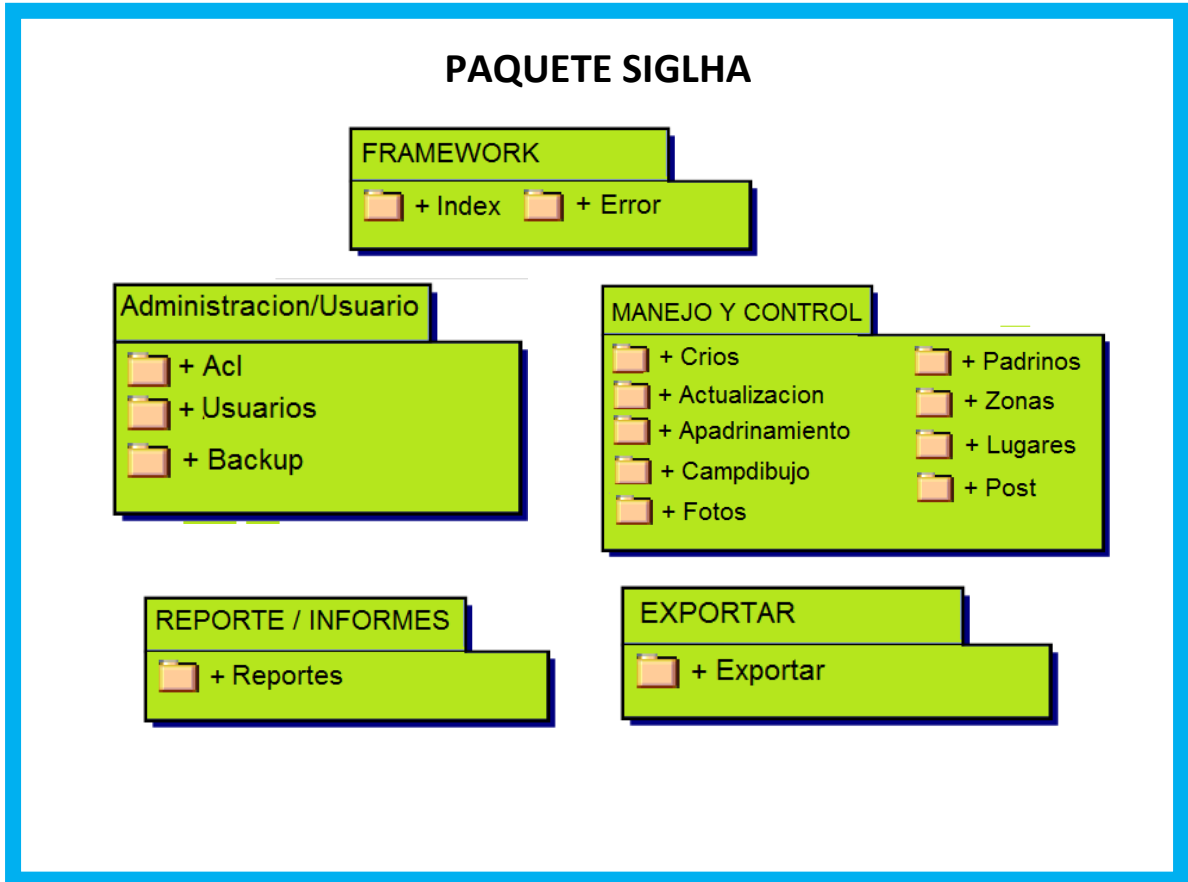


Figura 8: FRAMEWORK

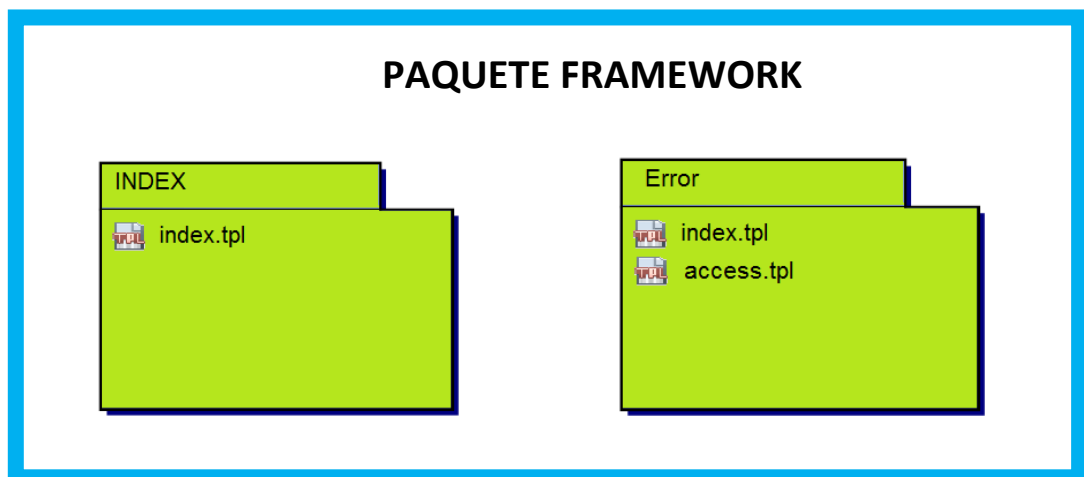


Figura 9: Administración/Usuario

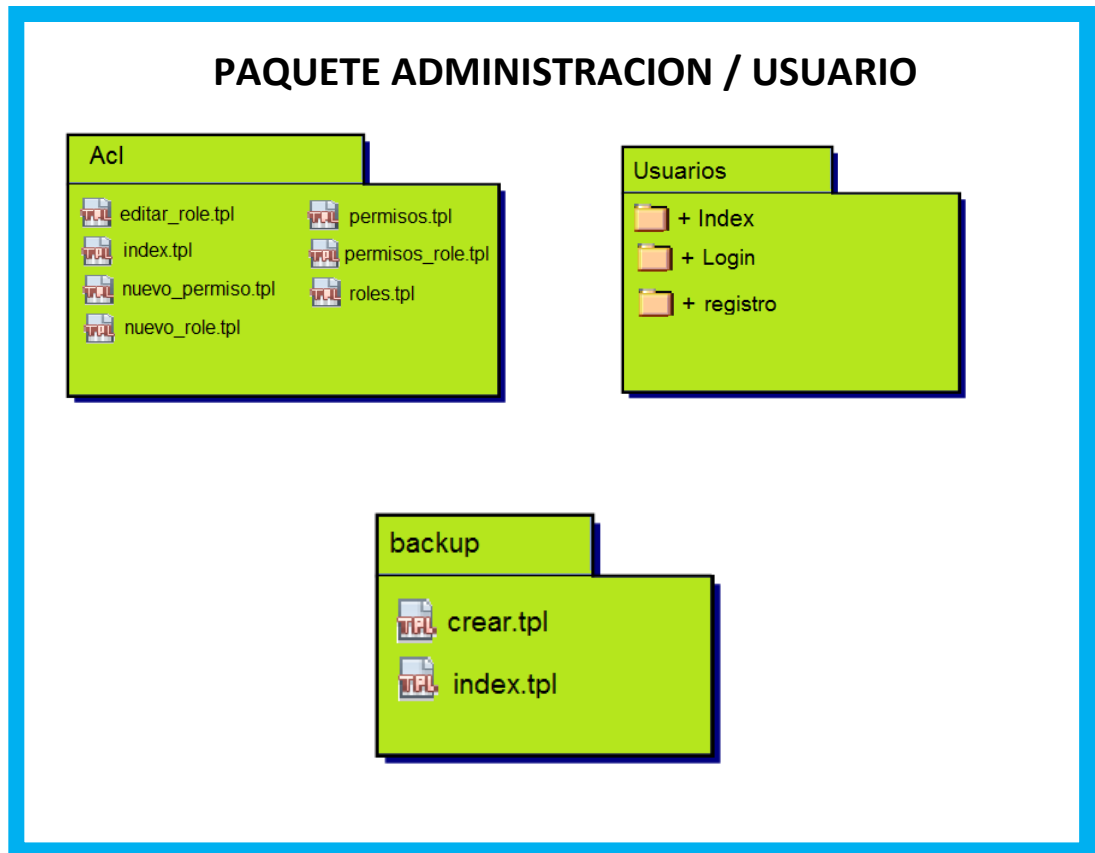




Figura 10: Usuarios

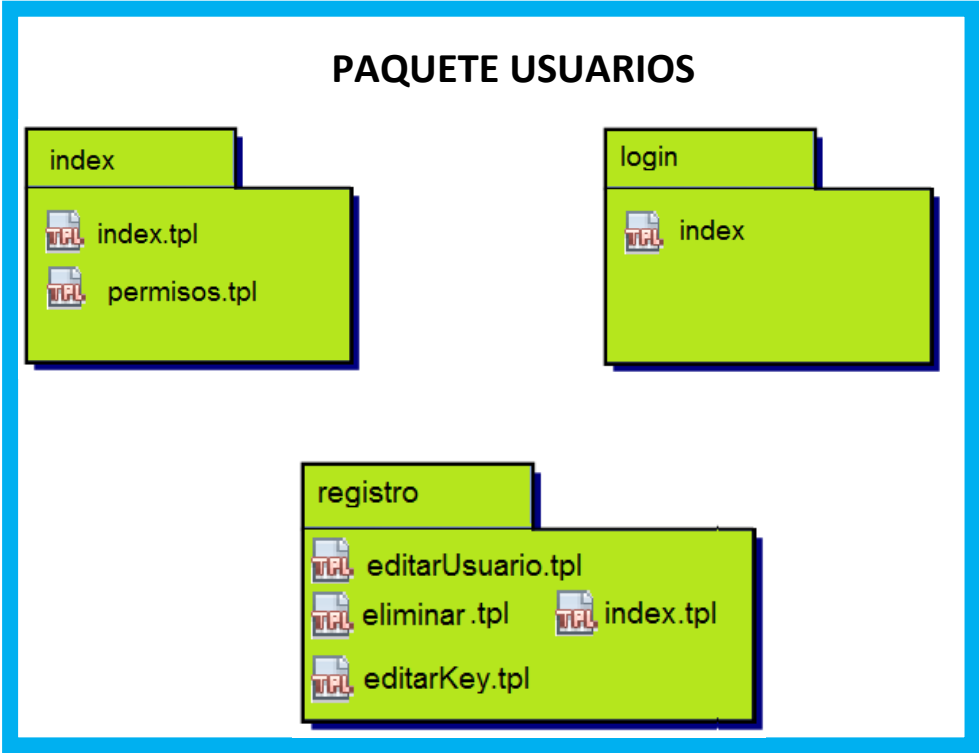


Figura 11: Manejo y control

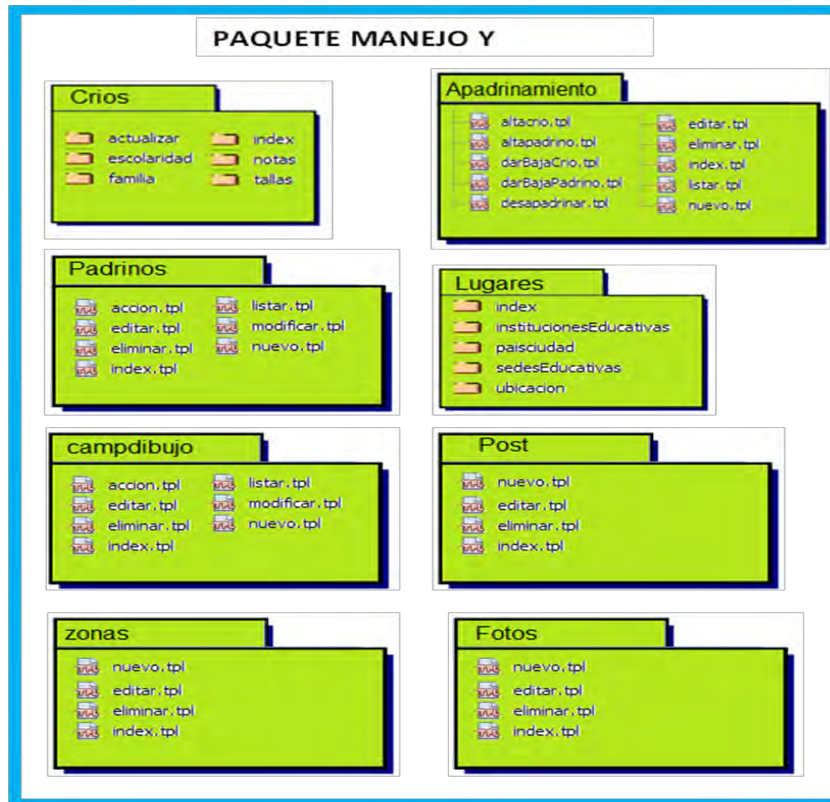
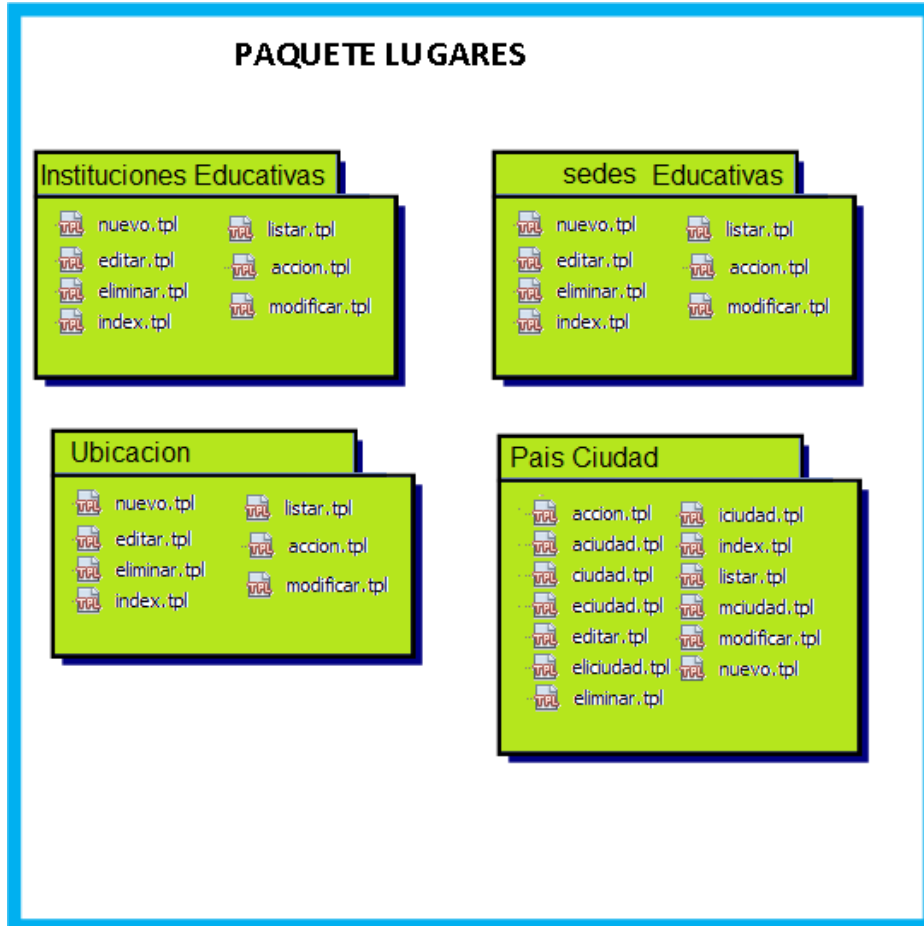


Figura 12  
Críos o Ahijados





**Figura 13**  
**Ubicación Lugares**

**Figura 14 Reportes e Informes**

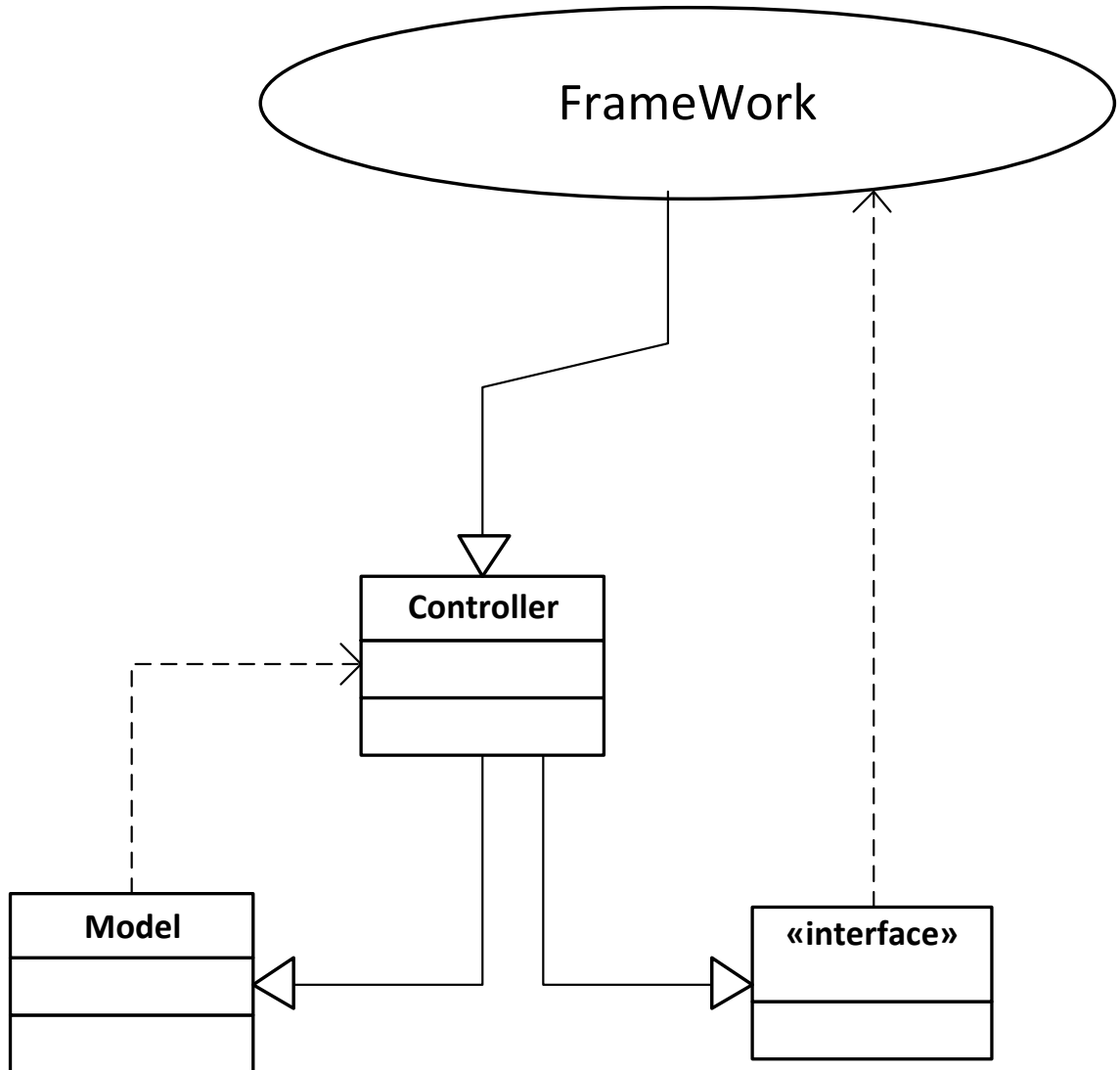


Figura 15: Exportar



## 9. DIAGRAMA DE CLASES

Figura 16: Diagrama de clases para el patrón MVC.<sup>7</sup>



<sup>7</sup> Para todas las clases e interfaces el diagrama es igual.

## 10. LISTADO DE CLASES

Figura 17: Clase Controlador principal del FRAMEWORK.

```
abstract class Controller
private $_registry;
protected $_view;
protected $_acl;
protected $_request;
public function __construct()
abstract public function index();
protected function loadModel($modelo, $modulo = false)
protected function getLibrary($libreria)
protected function getTexto($clave)
protected function getSoloTexto($clave)
protected function getFecha($clave)
protected function getInt($clave, $nocero=false, $limite=false)
protected function getIsInt($clave, $min=false, $max=false)
protected function getCadenaNumerica($clave)
protected function getCadenaNumerica($clave)
protected function redireccionar($ruta = false)
protected function filtrarInt($int)
protected function getPostParam($clave)
protected function getSql($clave)
protected function getAlphaNum($clave)
public function validarEmail($email)
protected function formatPermiso($clave)
public function reemplazaññ($txt, $contatenado=false)
```

Figura 18: Clase ERROR del FRAMEWORK.

```
class errorController extends Controller
public function __construct()
public function index()
public function access($codigo)
private function _getError($codigo = false)
```

Figura 19: Clases del módulo Administración /usuario

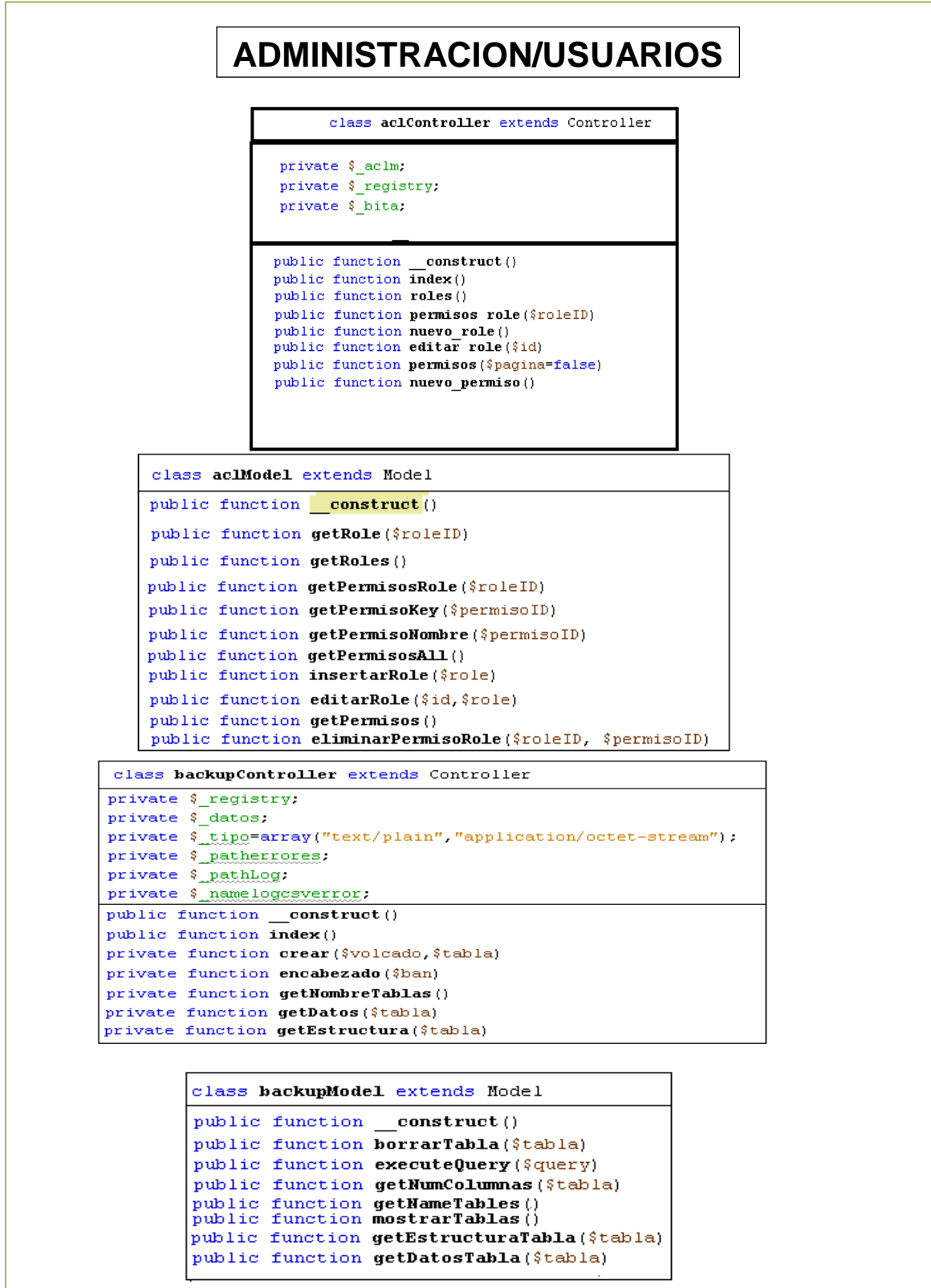


Figura 20: Clases del módulo Administración /usuario

## ADMINISTRACION/USUARIOS

```
class indexModel extends Model
public function __construct()
public function getUsuarios()
public function getUsuario($usuarioID)
public function getUsuarioID($usuarioID)
public function getPermisosUsuario($usuarioID)
public function getPermisosRole($usuarioID)
public function eliminarPermiso($usuarioID, $permisoID)
public function editarPermiso($usuarioID, $permisoID, $valor)
```

```
class registroModel extends Model
public function __construct()
public function verificarUsuario($usuario)
public function getRoles()
public function verificarEmail($email, $condicion="")
public function verificarEmailGlobal($email, $condicion="")
public function registrarUsuario
public function editarKeyUsuario($id, $pass)
public function editarUsuario
public function getUsuario($id)
public function activarUsuario($id)
public function eliminarUsuario($id)
```

```
class loginModel extends Model
public function __construct()
public function getUsuario($usuario, $password)
```

```
class loginController extends Controller
private $_login;
private $_registry;
private $_bita;
public function __construct()
public function index()
public function cerrar()
```



Figura 21: Clases del módulo Administración /usuario

## ADMINISTRACION/USUARIOS

```
class indexController extends usuariosController
private $_usuarios;
public function __construct()
public function index()
public function permisos($usuarioID)
```

```
class registroController extends Controller
private $_registro;
private $_registry;
private $ bita;
public function construct()
. public function index()
public function editarUsuario($id)
public function eliminar($id)
public function editarKey($id)
```

Figura 22: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class apadrinamientoController extends Controller

    private $_apadrinamiento;
    private $_registry;
    private $_bita;
    private $_crios;
    private $_padrinos;
    private $_escolaridad;

    public function __construct()
    public function getNombres()
    public function getPadrino()
    public function index()
    public function indexAjax()
    public function nuevo($id_nino=NULL)
    public function listar($pagina = false)
    public function eliminar($id)
    public function editar($id)
    public function desapadrinar()
    public function darBajaCrio()
    public function darBajaPadrino() _____
    public function altacrio()
    public function altapadrino()
```

```
class actualizarController extends Controller

    private $_registry;
    private $_bita;
    private $_crios;
    private $_actualizar;
    private $_escolaridad;
    private $_tallas;
    private $_familias;
    private $_notas;

    public function __construct()
    public function index()
    private function datosCompletos($array1,$fecha1,$dias_limite)
    private function diasEntreFechas($fechainicio, $fechafin)
    private function recomendaciones($tiempo,$limite)
```

```
class criosController extends Controller

    public function __construct()
    public function index()
```

Figura 23: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class fotosController extends Controller
private $_fotos;
private $_crios;
private $_rutaImagen;
private $_registry;
private $_bita;

public function __construct()
public function index($pagina = false)
public function indexAjax()
public function nuevo($codigo=NULL)
public function editar($codigo=false)
public function eliminar($id)
```

```
class campdibujoController extends Controller
private $_dibujos;

public function __construct()
public function index($pagina = false)
public function nuevo()
public function editar($id)
public function eliminar($id)
public function listar($pagina = false)
public function listarAjax()
public function modificar()
```

Figura 24: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class indexController extends Controller
private $_post;
public function __construct()
public function index()
```

```
class lugaresController extends Controller
public function __construct()
public function index()
```

```
class padrinosController extends Controller
private $_padrinos;
private $_registry;
private $_bita;
public function __construct()
public function index($pagina = false)
public function nuevo()
public function editar($id)
public function eliminar($id)
public function listar($pagina = false)
public function listarAjax()
public function modificar()
```

Figura 25: Clases del módulo Manejo y Control

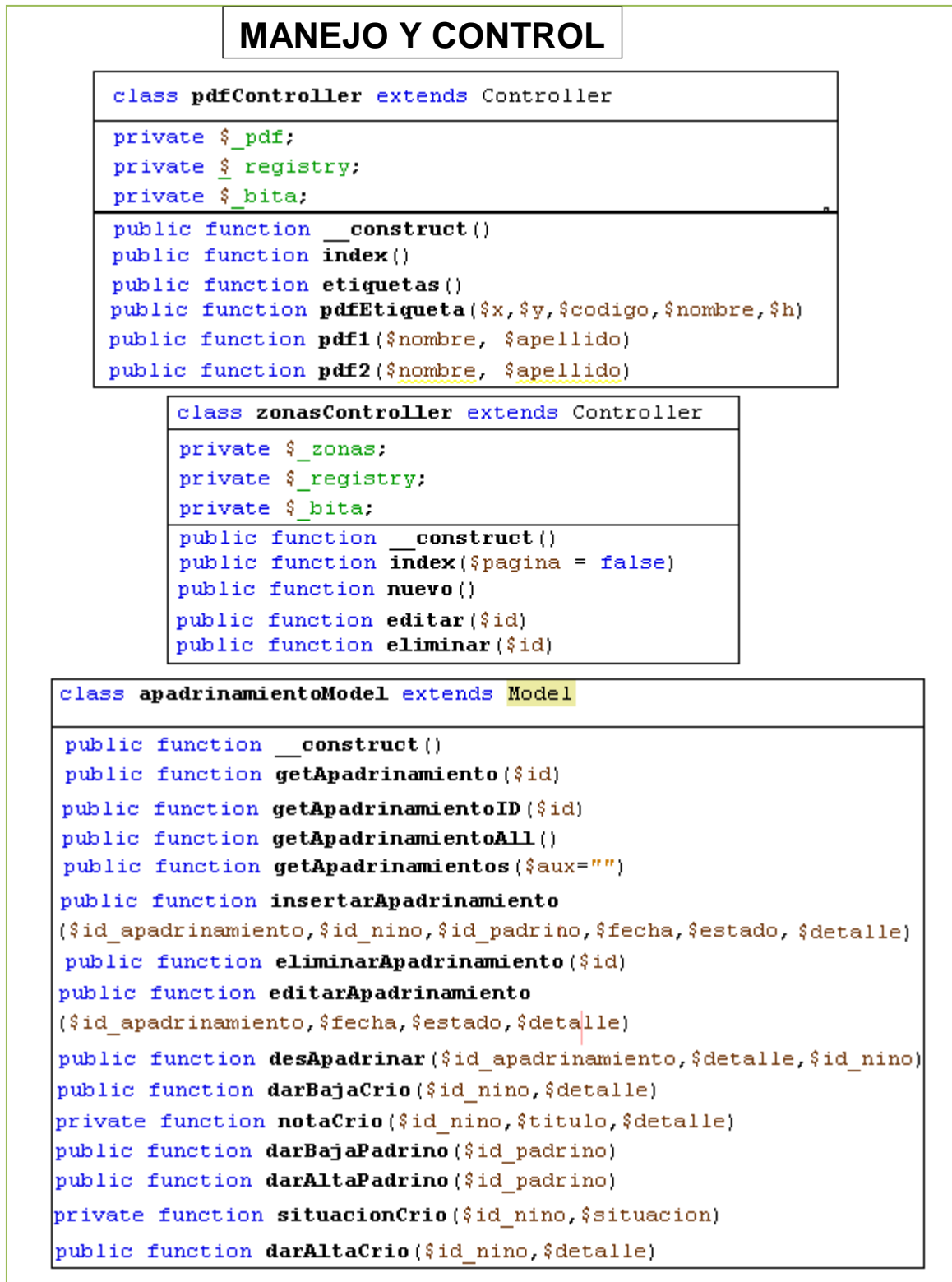


Figura 26: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class campdibujoModel extends Model
public function __construct()
public function getDibujos($condicion="")
public function getIdDibujos()
public function getCodigosFaltantes()
public function getCampDibujos($condicion="")
public function getCamp($id)
public function getCampana()
public function getDibujo($id)
public function getDibujoNino($id,$id_campana)
public function insertarDibujo($cn,$cc,$v1,$v2,$v3,$v4,$v5)
public function editarDibujo($id,$cc,$v1,$v2,$v3,$v4,$v5)
public function eliminarDibujo($id)
public function getUsuarios()
```

```
class fotosModel extends Model
public function __construct()
public function actualizarFoto($id,$fecha,$id_usuario)
public function getFotos()
public function getFotosUsuario($condicion="")
public function getFoto($id)
public function insertarFoto
($id_nino,$nombre,$fecha,$id_usuario)
public function editarFoto
($id_nino,$fecha,$id_usuario)
public function eliminarFoto($id)
public function getCodigosNinos()
public function getNinosConFotos()
public function getNinoConFoto($id)
public function getNino($id)
public function getUsuario($id)
public function getUsuarios()
```

Figura 27: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class padrinosModel extends Model

public function __construct ()
public function getPadrino ($id,$condicion='')
public function getPadrinos ($condicion='')
public function getPaises ()
public function insertarPadrino ( $id, $cc, $v1,$v2,$v3,$v4)
public function editarPadrino ($id,$cc, $v1,$v2,$v3,$v4,$v5)
public function eliminarPadrino ($id)
```

```
class postModel extends Model

public function __construct ()
public function getPosts ()
public function getPost ($id)
public function insertarPost ( $titulo, $cuerpo, $imagen)
public function editarPost ($id, $titulo, $cuerpo)
public function eliminarPost ($id)
public function insertarPrueba ($nombre)
public function getPrueba ($condicion = "")
public function seleccionar ($id)
public function deseleccionar ($id)
public function getPostSeleccionado ()
```

```
class zonasModel extends Model

public function __construct ()
public function getZonas ()
public function getZona ($id)
public function insertarZona
( $id, $id tecnico, $nombre)
public function editarZona
( $id, $id tecnico, $nombre)
public function eliminarZona ($id)
public function getTecnicos ()
```

Figura 28: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class actualizarController extends Controller
private $_registry;
private $_bita;
private $_crios;
private $_actualizar;
private $_escolaridad;
private $_tallas;
private $_familias;
private $_notas;
public function __construct()
public function index()
private function datosCompleto($array1,$fecha1,$dias_limite)
private function diasEntreFechas($fechainicio, $fechafin)
private function recomendaciones($tiempo,$limite)
```

```
class escolaridadController extends criosController
private $_escolaridad;
private $_registry;
private $_bita;
private $_crios;
private $_sede;
public function construct()
public function index()
public function indexAjax()
public function nuevo($id_nino=NULL)
private function grado($indice)
private function jornada($indice)
public function listar($pagina = false)
public function eliminar($id)
public function editar($id)
public function getNombres()
public function getSede()
```



Figura 29: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class familiaController extends Controller
private $_crios;
private $_registry;
private $_bita;
private $_familias;

public function __construct()
public function index()
public function indexAjax()
private function tipoDoc($tipo)
private function tipoParentesco($tipo)
public function nuevo($id_nino=NULL)
public function listar($pagina = false)
public function eliminar($id)
public function editar($id)
public function getNombres()
public function getCodigoFamiliar()
```

```
class indexController extends criosController
private $_crios;
private $_registry;
private $_bita;

public function __construct()
public function index()
public function indexAjax()
public function nuevo()
private function tipoDoc($tipo)
private function situacion($id)
private function motivos($id)
private function estados($id)
public function getUbicaciones()
public function getSedes()
public function listar($pagina = false)
public function eliminar($id)
public function editar($id)
```

Figura 30: Clases del módulo Manejo y Control

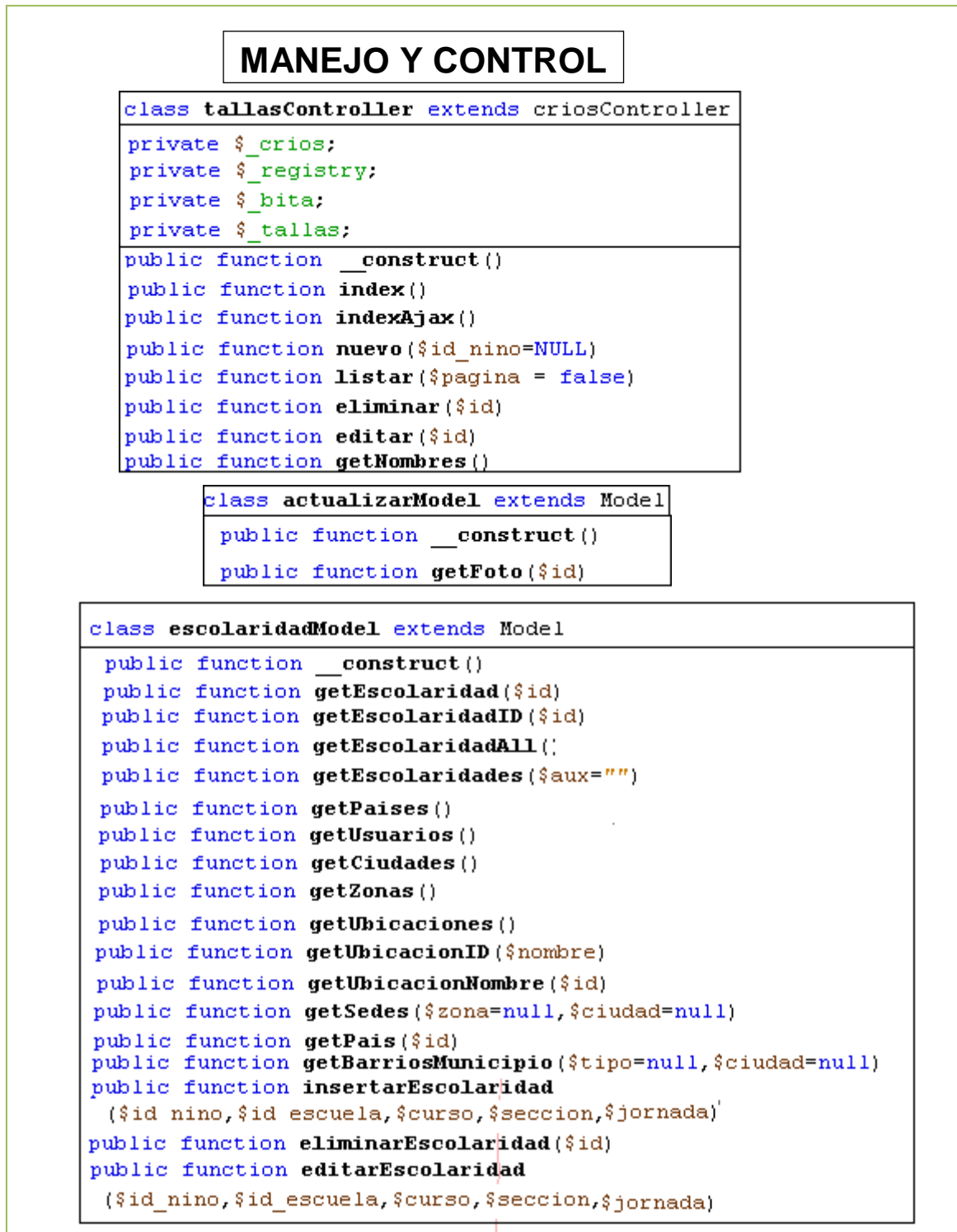


Figura 31: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class familiaModel extends Model

public function __construct ()
public function getFamilia ($id)
public function getMaxId ($id_nino)
public function getFamiliaID ($id)
public function getFamiliaAll ()
public function getFamilias ($aux="")
public function insertarFamilia
public function eliminarFamilia ($id)
public function editarFamilia
public function getCodigosNinos ()
```

```
class indexModel extends Model
public function __construct ()
public function getCrio ($id,$condicion="")
public function getCriosAlta ($condicion="")
public function getDatosCrio ($id)
public function getCrioID ($id)
public function getCriosAll ($condicion="")
public function getCrios ($aux="")
public function getSeguridad ()
public function getDeficiencia ()
public function getPaises ()
public function getUsuarios ()
public function getCiudades ()
public function getZonas ()
public function getUbicaciones ()
public function getUbicacionID ($nombre)
public function getUbicacionNombre ($id)
public function getSedes ($zona=null,$ciudad=null)
public function getPais ($id)
public function getBarriosMunicipio ($tipo=null,$ciudad=null)
public function insertarCrio ($id,$ref_crio,$tipo_doc,$documento,
public function eliminarCrio ($id)
public function editarCrio ($id,$ref_crio,$tipo_doc,$documento,
```

Figura 32: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class notasModel extends Model
```

```
public function __construct()  
public function getNota($cod_nota)  
public function getNotaID($id)  
public function getNotaAll()  
public function getNotas($aux="")  
  
public function insertarNota($id_nino,$titulo,$detalle)  
public function eliminarNota($id)  
public function editarNota($cod_nota,$id_nino,$titulo,$detalle)  
public function getTitulos()  
public function getCodigos()  
public function getCodigosNinos()
```

```
class tallasModel extends Model
```

```
public function __construct()  
public function getTalla($id)  
public function getTallaID($id)  
public function getTallaAll()  
public function getTallas($aux="")  
public function insertarTalla($id_nino,$camisa,$pantalon,$calzado,$fecha_toma)  
public function eliminarTalla($id)  
public function editarTalla($id_talla,$camisa,$pantalon,$calzado,$fecha)  
public function getTallaCamisas()  
public function getTallasPantalon()  
public function getCalzados()  
public function getCodigos()  
public function getCodigosNinos()
```

Figura 33: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class institucionesEducativasController extends Controller
    private $_ies;
    private $_municipio;
    private $_registry;
    private $_bita;

    public function __construct()
    public function index($pagina = false)
    public function nuevo()
    public function editar($id)
    public function modificar()
    public function eliminar($id)
    public function getCodCiudad()
    public function getCiudades()
    public function listar()
    public function listarAjax()
```

```
class sedesEducativasController extends Controller
    private $_sedes;
    private $_ubi;
    private $_ies;
    private $_registry;
    private $_bita;
    private $_municipio;

    public function __construct()
    public function index($pagina = false)
    public function nuevo()
    public function editar($id)
    public function getCodIes()
    public function getCodUbis()
    public function getCiudades()
    public function getPoblaciones()
    public function listar()
    public function listarAjax()
    public function eliminar($id)
    public function modificar()
```

Figura 34: Clases del módulo Manejo y Control

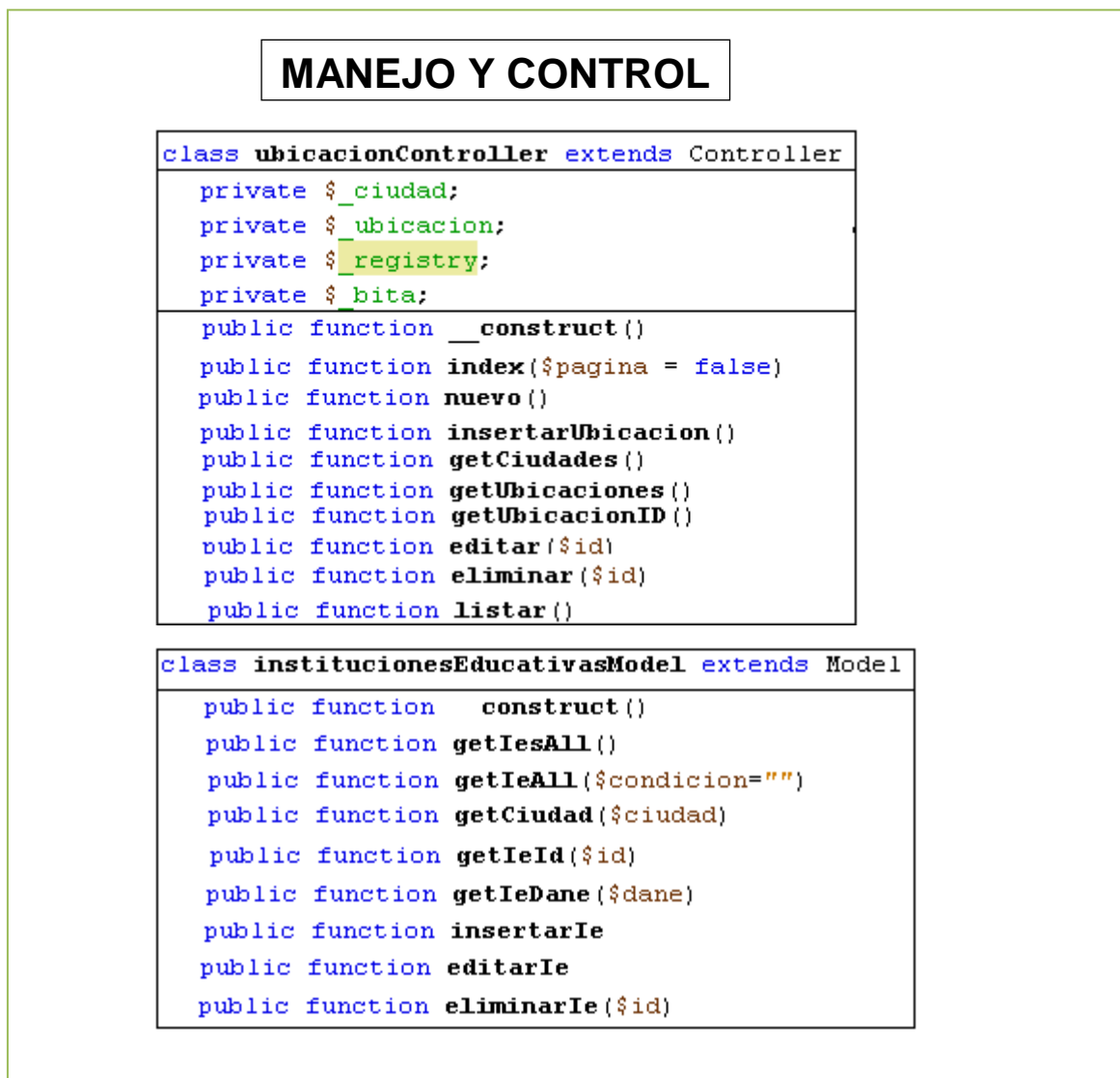


Figura 35: Clases del módulo Manejo y Control

## MANEJO Y CONTROL

```
class paisciudadModel extends Model
public function __construct()
public function getPaises()
public function getPais($id)
public function getCiudad($ciudad)
public function getCiudadId($id)
public function getCiudades($pais)
public function getCiudadesAll($condicion="")
public function insertarCiudad($id, $pais, $ciudad, $indtel)
public function insertarPais($cod, $nombre, $indtel)
public function editarPais($id, $nombre, $indtel)
public function editarCiudad($id, $nombre, $indtel)
public function eliminarPais($id)
public function eliminarCiudad($id)
```

```
class sedesEducativasModel extends Model
public function __construct()
parent::__construct();
public function getSedes()
public function getSedesGH()
public function getZonas()
public function getSede($id, $condicion="")
public function getSedesAll($condicion="")
public function getCodIe($nombre)
public function getCodUb($nombre)
public function insertarSede($cod_sede,
public function editarSede($cod_sede,
public function eliminarSede($id)
```

Figura 36: Clases del módulo Manejo y Control

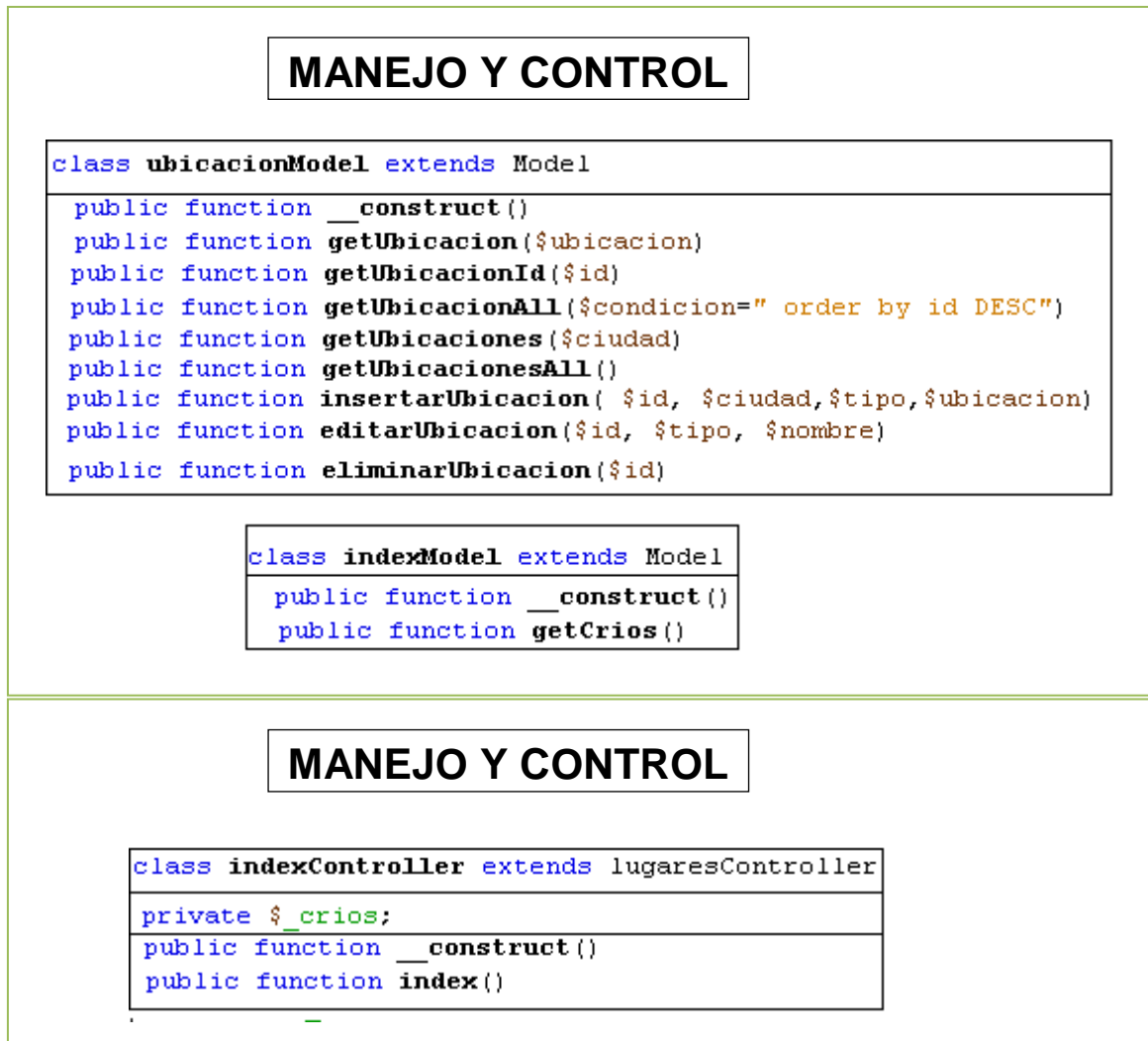




Figura 37: Clases del módulo Reportes e Informes

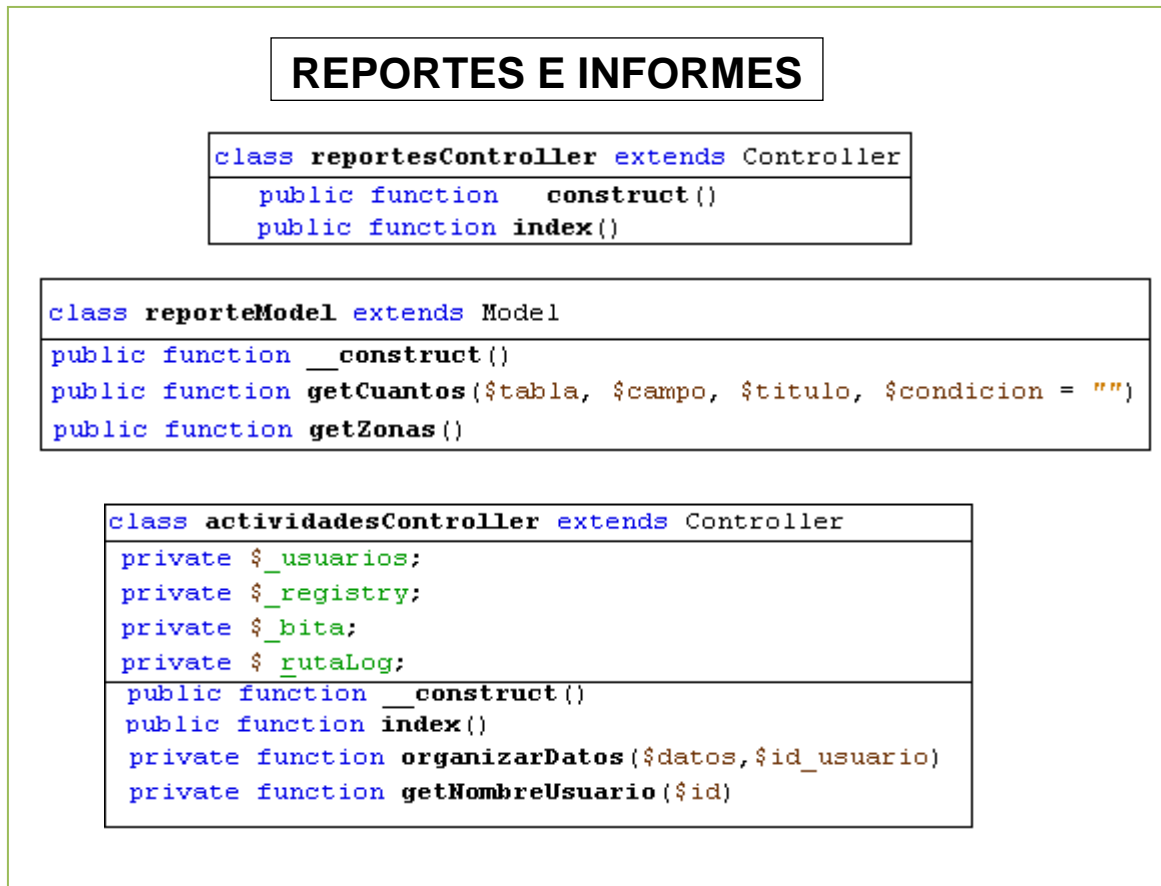


Figura 38: Clases del módulo Exportar

## EXPORTAR

```
class exportarController extends Controller
private $_expDat;
private $_registry;
private $ bita;
public function __construct ()
public function index ()
public function espanaAll ()
public function espanaBaja ()
public function espanaAlta ()
public function campDibujoExcel ()
private function getYavesLiterales (array $m)
private function Datos ($x=false);
```

```
class exportarModel extends Model
//private $cabeceras=array("");
public function __construct ()
public function getDatBogota ()
public function getCrios ($consulta=false)
public function getNotas ()
public function getNotasID ()
public function getDatCampDibujo ();
```

## 11. CONCLUSIONES

- Se logró que las sedes Bogotá, España y Tumaco trabajen en tiempo real haciendo más eficiente el trabajo de los funcionarios con relación a los tiempos dedicado a realizar actividades de actualización, entrega de información oportuna y veraz a los padrinos del estado de sus ahijados.
- Con SIGLHA implementado, la fundación empieza a realizar actividades de fidelización y no a gastar esfuerzos en actualizaciones, gracias a esto, las coordinaciones están planificando mejor las actividades y proyectos anuales al contar con más espacios y personal. Pensando incluso en expandir esta herramienta a los demás países donde hace presencia la fundación.
- El framework de SIGLHA sirve como base estructural de programación orientada a objetos en PHP y se convierte en una muy buena herramienta académica para futuros desarrollos orientados a la web.

## 12.RECOMENDACIONES

- Realizar copias de seguridad periódicamente
- Adquirir un servidor web para poder ampliar el sistema de información y poder adicionar los otros departamentos.
- Cambiar de contraseñas periódicamente.
- Realizar retroalimentación con los usuarios para mejorar el sistema.
- Promover el framework creado para SIGLHA para que sea usado y mejorado académicamente por los compañeros que empiezan a desarrollar sus proyectos orientados a la web y poder contar con un framework universitario propio.

### 13. BIBLIOGRAFÍA

- BASCON, E. . El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing. Acta Nova, 2004.
- BECK, KENT. Una Explicación de la Programación Extrema: Aceptar el cambio. Madrid: Addison-Wesley Iberoamericana Espanya, S.A. Todo el libro, 2002.
- CALVACHE BENAVIDES, Maria Fernanda. Sistema de información y documentación en la sección de talento humano EMPOPASTO S.A.E.S.P 2004. <http://biblioteca.udenar.edu.co>
- CREACIÓN DEL FRAMEWORK BÁSICO <http://www.dlancedu.com>, usado como base estructural para la creación de SIGLHA (último acceso: 2 de Abril de 2013).
- FUNDACION GLOBAL HUMANITARIA COLOMBIA. Que somos y que hacemos [http://www.globalhumanitariacolombia.org/que\\_hace\\_global\\_humanitaria\\_en\\_colombia](http://www.globalhumanitariacolombia.org/que_hace_global_humanitaria_en_colombia) (último acceso: 1 de Abril de 2013).
- GARZÓN URBANO, Maribel, ORTIZ VELA, Martha Isabel y URBANO CALVACHE, Laura Inés. Sistema de información en un ambiente web para la administración y control de la información que relaciona los entes secretaría de educación municipal e instituciones educativas de Pasto, Elaborado por los Ingenieros, Y en 2003. <http://biblioteca.udenar.edu.co>
- KENDALL Kenneth E. Kendall Julie E. Análisis y Diseño de Sistemas, 3ª Edición, KENDALL & KENDALL, México, 1995.
- LENGUAJE DE PROGRAMACION PHP. <http://es.wikipedia.org/wiki/PHP> (último acceso: 3 de Abril de 2013).
- MANUAL BASICOACCES 2007. <http://www.proulex.com/computo/oe/doc/basico-access2007.pdf> (último acceso: 2 de Abril de 2013).
- MANUAL MYSQL PARA OPTIMIZAR LA IMPLEMENTACIÓN DE LA BASE DE DATOS.<http://dev.mysql.com/doc/refman/5.0/es/>, (último acceso: 2 de Abril de 2013).
- MANUAL PHP PARA OPTIMIZAR EL PROCESO DE PROGRAMACIÓN <http://php.net/manual/es/index.php>, (último acceso: 2 de Abril de 2013).
- MONTES SOTO, Julián Andrés y ROSERO MONTÚFAR, Edmundo Javier. Sistema de información bibliotecario liceo de la universidad de Nariño, 2008.<http://biblioteca.udenar.edu.co>
- POWELL Thomas A. HTML & CSS: The Complete Reference, Fifth Edition, McGraw-Hill, 2011.

PROGRAMACION EXTREMA. Recuperado el '1 de diciembre de 2010, de Programación Extrema. Disponible internet: <http://www.programacionextrema.org/>