

**UNIVERSIDAD ALAS PERUANAS**  
**FACULTAD DE INGENIERÍAS Y ARQUITECTURA**  
ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA



**TESIS**  
**IMPLEMENTACIÓN DE UN SISTEMA**  
**INFORMÁTICO DE GESTIÓN PARA EL CONTROL**  
**DE SACRAMENTOS Y ASIGNACIÓN DE**  
**PERSONAL PARA LA PARROQUIA SAN**  
**CLEMENTE DE PISCO**

**PRESENTADO POR EL ALUMNO**

**JUAN GUILLERMO CASTILLO HERNÁNDEZ**

**PARA OPTAR EL TÍTULO PROFESIONAL DE**  
**INGENIERO DE SISTEMAS E INFORMÁTICA**

**PISCO – PERÚ**

**2013**

## **DEDICATORIA**

*A mi familia, por darme la oportunidad de ser  
alguien exitoso, y a mis compañeros, por sacar  
lo mejor de mí y poder seguir adelante.*

## **AGRADECIMIENTOS**

*A mi familia, por darme todo el apoyo que necesitaba.*

*A mis compañeros, por compartir distintas  
experiencias en todos estos años.*

*A la parroquia San Clemente, por formarme espiritualmente y  
permitirme realizar este proyecto.*

*A mis profesores, por compartir todo su conocimiento  
y experiencia propia para que seamos los mejores.*

*A mi tutor, Luis Loo Paríán, por su guía y exigencia para realizar  
un gran trabajo, fruto de mi propio esfuerzo.*

## **RESUMEN**

En el inicio del proyecto se puede observar, mediante el “Acta de Constitución del Proyecto”, el control manual y engorroso de los sacramentos y del personal de la parroquia. Los tiempos que se toman para brindar el servicio adecuado a la feligresía sobrepasan con facilidad las 24 horas, haciendo que se tenga que hacer un gasto extra de dinero y tiempo, que fácilmente podría evitarse, así como el saber con certeza que personal está asociado a que comunidad dentro de la parroquia, perdiendo así la capacidad de poder distribuir mejor su recurso humano.

Durante la planificación, se estima que el proyecto tendrá un costo de 13,065.80 nuevos soles, cantidad que contiene los costos del equipo que desarrollará el proyecto así como de los materiales, equipos y servicios que se tendrán que utilizar para poder llevar a cabo con éxito el proyecto. Se expone que el proyecto tendrá una duración de 8 meses, empezando el 19 de Abril del 2013 y terminando el 21 de Noviembre del mismo año. Se establecen los tiempos de duración de cada actividad así como el recurso humano responsable de cada uno de ellos. Junto con esto, las fechas en las cuales se entregaran avances del proyecto que permitan al cliente validar cada parte del software elaborado.

En la ejecución, se define a “N-Capas” como arquitectura a utilizar para el desarrollo del software, también el uso de la metodología de desarrollo “RUP” de IBM para dar soporte al modo en que el software es construido y probado. Se validan también los costos que se planificaron y se determina que los costos se encuentran dentro del rango aceptable del presupuesto que se hizo en la planificación.

Durante el seguimiento y control, se describen los problemas que surgieron frente a los cambios de fechas de entrega y requerimientos iniciales que se establecieron antes de empezar el desarrollo del software. Se tuvieron que actualizar algunos documentos y cambiar las fechas de las actividades afectadas por los cambios. Por fortuna, se pudo completar con la nueva programación de fechas, aunque llevo su tiempo y dedicación el hacerlo.

Por último, el cierre del proyecto muestra que el cliente queda satisfecho con el producto entregado, puesto que este cumple con todos los requerimientos que se pidieron, quedando como evidencia actas e informes firmados por el párroco, sponsor del proyecto y la persona que valida que efectivamente se cumplió con el proyecto en términos de requerimientos, costos, tiempos y calidad.

## INTRODUCCIÓN

La presente tesis consiste en el desarrollo e implantación de un sistema software que permita llevar un mejor control de los sacramentos que reciben las personas en la parroquia San Clemente de Pisco y llevar un registro de las personas que conforman las distintas comunidades o grupos parroquiales que forman parte de la parroquia. El proyecto de implementación del software se llevará a cabo teniendo como base las buenas prácticas de gestión de proyectos del PMBOK, de forma que se utilicen las 5 fases de desarrollo de proyecto: inicio, planificación, ejecución, seguimiento y control y por último el cierre.

Durante el inicio, se presentará el resultado del análisis de la problemática de la parroquia, así como la justificación e importancia del desarrollo del proyecto y como se beneficiaría la parroquia con este.

En la planificación, se encontrará la información sobre el estudio que se siguió para poder realizar el proyecto, el alcance que tendrá el proyecto, los tiempos y actividades que comprende el desarrollo del software, los costos asociados a los recursos que se utilizarán, la organización del equipo de desarrollo y sus funciones, la dirección que tomará la comunicación de información del equipo de proyecto al cliente, los riesgos a los que podría enfrentarse el proyecto y las adquisiciones de los recursos que tendrán que llevarse a cabo.

Durante la ejecución, se pondrá énfasis al desarrollo del software propiamente dicho, y se define la metodología de desarrollo a usar, así como la arquitectura que se utilizará. Se expondrá también el esquema que se siguió para poder desarrollar el sistema, contando con su arquitectura, diseño, código fuente y manuales del mismo. Durante la gestión del proyecto de esta fase, se contará con la documentación sobre actas firmadas por el cliente, lo cual queda como evidencia que el proyecto se está desarrollando de acuerdo a las fechas programadas.

En el seguimiento y control, se podrá observar el modo en que el software está siendo validado y controlado en términos de fechas y calidad. También se visualizarán los distintos problemas a los cuáles el proyecto se enfrenta y que acciones se tomaron frente a estos para no detener el desarrollo del mismo.

En el cierre del proyecto, se encontrarán las lecciones aprendidas durante el desarrollo del proyecto y los cuáles se tomarán en cuenta para un futuro proyecto que se desarrolle, además de contar con la respectiva documentación de conformidad en la que el cliente expresa su satisfacción con respecto al software que se le fue entregado. También se toma en cuenta la capacitación del usuario para que este pueda usar sin problemas el software y el modo como este fue instalado en la parroquia.

Por último, se colocarán las conclusiones a las que se llegaron después de haber desarrollado el proyecto y que recomendaciones se proponen.

## ÍNDICE PRINCIPAL

CAPÍTULO I: ANÁLISIS DE LA ORGANIZACIÓN .....	17
1.1. Fines de la Organización .....	18
1.1.1. Visión.....	18
1.1.2. Misión .....	18
1.1.3. Valores .....	18
1.1.4. Objetivos Estratégicos.....	18
1.1.5. Unidades Estratégicas de Negocios .....	18
1.2. Análisis externo .....	19
1.2.1. Análisis del entorno general .....	19
A. Factores económicos.....	19
B. Factores tecnológicos .....	19
C. Factores políticos.....	19
D. Factores sociales .....	19
E. Factores demográficos .....	20
1.3. Análisis Interno .....	20
1.3.1. Recursos y capacidades .....	20
A. Recursos tangibles .....	20
B. Recursos intangibles .....	20
C. Capacidades organizativas.....	20
D. Análisis de recursos y capacidades .....	21
1.3.2. Análisis de la cadena de valor .....	21
A. Actividades primarias .....	21
B. Actividades de apoyo.....	21
1.4. Análisis Estratégico .....	21
1.4.1. Análisis FODA .....	21
A. Fortalezas .....	21
B. Oportunidades .....	22
C. Debilidades .....	22
D. Amenazas .....	22
1.4.2. Matriz FODA .....	23
CAPÍTULO II: MARCO TEÓRICO DEL NEGOCIO Y DEL PROYECTO .....	24
2.1. Marco teórico del Negocio .....	25
2.1.1. Marco teórico del Negocio 1.....	25

<b>2.1.2. Marco teórico del Negocio 2.....</b>	26
<b>2.2. Marco teórico del Proyecto.....</b>	26
<b>2.2.1. Gestión del Proyecto .....</b>	26
<b>2.2.2. Ingeniería del Proyecto .....</b>	30
<b>2.2.3. Soporte del Proyecto.....</b>	34
<b>CAPÍTULO III: INICIO Y PLANIFICACIÓN DEL PROYECTO .....</b>	36
<b>3.1. Gestión del proyecto .....</b>	37
<b>3.1.1. Iniciación .....</b>	37
<b>A. Nacimiento del proyecto .....</b>	37
<b>B. Justificación .....</b>	37
<b>C. Importancia .....</b>	37
<b>D. Acta de constitución del proyecto.....</b>	38
<b>E. Identificación de los interesados .....</b>	44
<b>3.1.2. Planificación .....</b>	44
<b>A. Alcance .....</b>	44
<b>Definición del alcance del proyecto y alcance del producto.....</b>	44
<b>Requerimientos.....</b>	45
<b>EDT.....</b>	48
<b>Entregables.....</b>	51
<b>B. Tiempo.....</b>	52
<b>Definición de las Actividades.....</b>	52
<b>Hitos.....</b>	56
<b>Cuadro de los responsables de las tareas .....</b>	57
<b>Cronograma de actividades .....</b>	60
<b>C. Costo .....</b>	65
<b>Línea Base.....</b>	65
<b>Identificación de recursos .....</b>	65
<b>Plan de Gestión para Costos .....</b>	66
<b>Cuadro de Costos .....</b>	68
<b>Presupuesto.....</b>	71
<b>D. Recursos Humanos .....</b>	76
<b>Organización .....</b>	76
<b>Matriz de asignación de responsabilidades (RAM).....</b>	77
<b>Plan de Gestión del Personal.....</b>	81

<b>E. Comunicaciones</b> .....	83
<b>Directorio de Stakeholders</b> .....	83
<b>Plan de Comunicaciones</b> .....	84
<b>F. Riesgos</b> .....	96
<b>Listado de Riesgos</b> .....	96
<b>Identificación, Estimación y Priorización de Riesgos</b> .....	97
<b>Documento de Análisis de Riesgos del Proyecto</b> .....	100
<b>G. Adquisiciones</b> .....	103
<b>Infraestructura, Equipos, Materiales y Accesorios</b> .....	103
<b>Plan de Adquisiciones</b> .....	104
<b>3.2. Ingeniería del proyecto</b> .....	115
<b>3.2.1. Modelamiento de requerimientos</b> .....	115
<b>3.2.2. Diseño</b> .....	122
<b>3.3. Soporte del proyecto</b> .....	125
<b>3.3.1. Planificación de la Calidad</b> .....	125
<b>3.3.2. Identificación de estándares y métricas</b> .....	127
<b>3.3.3. Diseño de formatos de aseguramiento de calidad</b> .....	129
<b>CAPÍTULO IV: EJECUCIÓN Y SEGUIMIENTO Y CONTROL DEL PROYECTO</b> .....	132
<b>4.1. Gestión del proyecto</b> .....	133
<b>4.1.1. Ejecución</b> .....	133
<b>A. Acta de aceptación de entregables a aprobar</b> .....	133
<b>B. Acta de reunión de equipo</b> .....	136
<b>C. Cronograma del proyecto</b> .....	143
<b>D. Hoja de costos</b> .....	148
<b>4.1.2. Seguimiento y control</b> .....	151
<b>A. Matriz de trazabilidad de requerimientos</b> .....	151
<b>B. Estructura de la descomposición de trabajo WBS</b> .....	156
<b>C. Informe de estado externo</b> .....	159
<b>D. Riesgos del proyecto</b> .....	164
<b>E. Solicitud de cambios</b> .....	165
<b>4.2. Ingeniería del proyecto</b> .....	168
<b>A. Concepción (Incepción)</b> .....	168
Especificación de requerimientos de software.....	168
Modelos de caso de uso del sistema .....	174

<b>B. Elaboración .....</b>	178
Diseño del sistema y arquitectura del software .....	178
Modelo de datos .....	191
<b>C. Construcción .....</b>	192
Entorno de construcción, desarrollo y pruebas preparado .....	192
Construcción de software y pruebas unitarias.....	195
Integración de software y pruebas de integración.....	365
Manuales .....	367
<b>D. Transición.....</b>	384
Pruebas de aceptación del cliente.....	384
Despliegue del software .....	396
Plan de capacitación .....	396
<b>4.3. Soporte del proyecto .....</b>	398
<b>4.3.1. Gestión de la configuración .....</b>	398
<b>4.3.2. Aseguramiento de la calidad.....</b>	401
<b>4.3.3. Medición del valor ganado.....</b>	416
<b>4.3.4. Métricas y evaluación del desempeño .....</b>	424
<b>CAPÍTULO V: CIERRE DEL PROYECTO.....</b>	434
<b>5.1. Gestión del proyecto .....</b>	435
<b>5.1.1. Cierre .....</b>	435
A. Lecciones Aprendidas .....	435
B. Acta de Reunión de Cierre.....	440
C. Certificado de Conformidad .....	441
<b>5.2. Ingeniería del proyecto .....</b>	442
<b>5.2.1. Pruebas .....</b>	442
A. Plan de Pruebas.....	442
B. Informe de pruebas de Aceptación .....	443
<b>5.2.2. Implementación .....</b>	444
A. Plan de capacitación .....	444
B. Manual de Usuario .....	446
<b>5.3. Soporte del proyecto .....</b>	525
<b>5.3.1. Gestión de la configuración .....</b>	525
<b>5.3.2. Aseguramiento de la calidad.....</b>	527
<b>5.3.3. Métricas y evaluación del desempeño .....</b>	529

<b>CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES.....</b>	532
<b>    6.1. Conclusiones.....</b>	533
<b>    6.2. Recomendaciones.....</b>	533
<b>GLOSARIO DE TÉRMINOS .....</b>	534
<b>BIBLIOGRAFÍA.....</b>	536

## ÍNDICE DE GRÁFICOS

Ilustración 1 Cuadro de Grupos de Procesos y Área de Conocimiento .....	29
Ilustración 2 Arquitectura General de RUP .....	30
Ilustración 3 Diseño Básico de Capas .....	33
Ilustración 4 Organigrama del Equipo de Proyecto .....	76
Ilustración 5 Diagrama de Componentes.....	122
Ilustración 6 Diagrama de Despliegue.....	122
Ilustración 7 Diagrama de Paquetes .....	123
Ilustración 8 Diagrama de Secuencia de Asignación .....	123
Ilustración 9 Diagrama de Secuencia de Sacramentos .....	124
Ilustración 10 Caso de Uso de Feligrés.....	174
Ilustración 11 Caso de Uso de Templos .....	174
Ilustración 12 Caso de uso del Libro Sacramental.....	175
Ilustración 13 Caso de Uso de Sacramentos .....	175
Ilustración 14 Caso de Uso de Personal .....	176
Ilustración 15 Caso de Uso de Comunidades.....	176
Ilustración 16 Caso de Uso de Asignación.....	177
Ilustración 17 Diagrama de Paquetes .....	177
Ilustración 18 Prototipo del menú Básico .....	178
Ilustración 19 Prototipo del menú Control de Sacramentos .....	178
Ilustración 20 Prototipo del menú Asignación de Personal .....	178
Ilustración 21 Prototipo de la interfaz de Países.....	179
Ilustración 22 Prototipo de la interfaz de Ciudades.....	179
Ilustración 23 Prototipo de la interfaz de Ciudades.....	180
Ilustración 24 Prototipo de la interfaz de Tipos .....	180
Ilustración 25 Prototipo de la interfaz de Lugares de Reunión.....	181
Ilustración 26 Prototipo de la interfaz de Salvar Datos.....	181
Ilustración 27 Prototipo de la interfaz de Restaurar Datos.....	181
Ilustración 28 Prototipo de la interfaz de Feligrés 1 .....	182
Ilustración 29 Prototipo de la interfaz de Feligrés 2 .....	182
Ilustración 30 Prototipo de la interfaz del Libro Sacramental .....	183
Ilustración 31 Prototipo de la interfaz de Templos.....	183
Ilustración 32 Prototipo de la interfaz de Bautismos 1.....	184
Ilustración 33 Prototipo de la interfaz de Bautismos 2.....	184
Ilustración 34 Prototipo de la interfaz de Confirmaciones 1 .....	185
Ilustración 35 Prototipo de la interfaz de Confirmaciones 2 .....	185
Ilustración 36 Prototipo de la interfaz de Matrimonios 1.....	186
Ilustración 37 Prototipo de la interfaz de Matrimonios 2.....	186
Ilustración 38 Prototipo de la interfaz de Matrimonios 3.....	187
Ilustración 39 Prototipo de la interfaz de Actas de Sacramentos.....	187
Ilustración 40 Prototipo de la interfaz de Personal 1.....	188
Ilustración 41 Prototipo de la interfaz de Personal 2.....	188
Ilustración 42 Prototipo de la interfaz de Comunidades .....	189

Ilustración 43 Prototipo de la interfaz de Asignaciones.....	189
Ilustración 44 Prototipo de la interfaz de Personal Asignado.....	190
Ilustración 45 Diagrama de Clases .....	191
Ilustración 46 Capas del Sistema.....	192
Ilustración 47 Estructura de la Capa Entidades.....	192
Ilustración 48 Estructura de la Capa AccesoDatosNegocio.....	193
Ilustración 49 Estructura de la Capa Presentación .....	194
Ilustración 50 Diagrama de Componentes.....	195
Ilustración 51 Diagrama de Despliegue.....	365
Ilustración 52 Diagrama de Despliegue.....	396
Ilustración 53 Gráfico de Valor Ganado .....	423

## ÍNDICE DE TABLAS

Tabla 1 Matriz FODA .....	23
Tabla 2 Requerimientos del Proyecto .....	47
Tabla 3 Definición de las actividades .....	56
Tabla 4 Hitos del Proyecto .....	56
Tabla 5 Cuadro de los responsables de las tareas.....	59
Tabla 6 Plan de Gestión de Costos .....	67
Tabla 7 Costos de Materiales .....	68
Tabla 8 Costos de Equipos.....	69
Tabla 9 Costos de Servicios .....	69
Tabla 10 Costos del Recurso Humano.....	70
Tabla 11 Resumen de Costos Totales.....	70
Tabla 12 Presupuesto Resumen del Proyecto.....	75
Tabla 13 Matriz RAM.....	80
Tabla 14 Directorio de Stakeholders .....	83
Tabla 15 Listado de Riesgos .....	96
Tabla 16 Matriz de Riesgos.....	99
Tabla 17 Cuadro de Análisis de Riesgos .....	102
Tabla 18 Requerimientos del Producto.....	121
Tabla 19 Métrica de Calidad del Producto .....	127
Tabla 20 Métrica de Calidad del Proyecto .....	128
Tabla 21 Formato de Pruebas Unitarias.....	131
Tabla 22 Hoja de Costos .....	150
Tabla 23 Matriz de Trazabilidad de Requerimientos .....	155
Tabla 24 Tabla de Riesgos del Proyecto .....	164
Tabla 25 Especificación de Requerimientos de Software .....	173
Tabla 26 Caso de Prueba de Gestión de Feligrés .....	354
Tabla 27 Caso de Prueba de Gestión de Sacramentos.....	357
Tabla 28 Caso de Prueba de Gestión de Libro Sacramental.....	359
Tabla 29 Caso de Prueba de Gestión de Templos .....	360
Tabla 30 Caso de Prueba de Gestión de Personal.....	362
Tabla 31 Caso de Prueba de Gestión de Comunidades.....	364
Tabla 32 Caso de Prueba de Gestión de Asignación .....	365
Tabla 33 Plan de Pruebas Integrales .....	366
Tabla 34 Plan de Capacitación de Usuarios.....	397
Tabla 35 Pruebas Unitarias del Módulo de Feligrés.....	403
Tabla 36 Pruebas Unitarias del Módulo de Sacramentos y Reportes de Sacramentos .....	406
Tabla 37 Pruebas Unitarias del Módulo de Personal .....	409
Tabla 38 Pruebas Unitarias del Módulo de Comunidades .....	412
Tabla 39 Pruebas Unitarias del Módulo de Reportes de Asignación y Backup's .....	415
Tabla 40 Cuadro de Control de Avance .....	418
Tabla 41 Cuadro de Costo Real .....	422
Tabla 42 Tabla Resumen de Presupuesto, Valor Ganado y Costo Real.....	422

Tabla 43 Estado de Avance del Proyecto Nº 1 .....	426
Tabla 44 Estado de Avance del Proyecto Nº 2 .....	428
Tabla 45 Estado de Avance del Proyecto Nº 3 .....	430
Tabla 46 Estado de Avance del Proyecto Nº 4 .....	433
Tabla 48 Plan de Pruebas .....	442
Tabla 49 Plan de Capacitación.....	445
Tabla 50 Evaluación del Desempeño .....	531

## **ÍNDICE DE FORMATOS DE LA GESTIÓN DEL PROYECTO**

Acta de Constitución .....	538
Declaración del Alcance .....	540
Plan de Gestión de Riesgos .....	541
Plan de Gestión de Adquisiciones .....	545
Plan de Gestión de Calidad .....	547
Plan de Gestión de Recursos Humanos.....	548
Plan de Gestión de Comunicaciones .....	549

## **ÍNDICE DE ANEXOS**

Anexo Nº 1: Declaración del Alcance .....	552
Anexo Nº 2: Plan de Gestión de Riesgos .....	556
Anexo Nº 3: Constancia de Bautismo.....	567
Anexo Nº 4: Constancia de Confirmación .....	568
Anexo Nº 5: Constancia de Matrimonio .....	569

## **CAPÍTULO I: ANÁLISIS DE LA ORGANIZACIÓN**

## **1.1. Fines de la Organización**

### **1.1.1. Visión**

Ser una comunidad de comunidades que lleve a todos los fieles el anuncio del evangelio encomendado por Cristo, para construir una comunidad de fe, de culto, de amor y de servicio fraterno.

### **1.1.2. Misión**

Consolidar nuestra parroquia como una comunidad misionera a partir de una formación integral que lleve el evangelio a familias, jóvenes, pobres y alejados para instaurar el reino de Dios.

### **1.1.3. Valores**

- Solidaridad
- La Organización como Instrumento
- La Superación Personal
- Dar buen testimonio de vida
- Voluntad para el servicio pastoral
- Practicamos el amor de Dios a través de nuestros hermanos
- Trabajar en forma consciente y activa
- La comunicación en la familia
- Responsabilidad sobre cada una de nuestras acciones
- Todo evangelizado debe ser un agente de cambio

### **1.1.4. Objetivos Estratégicos**

- Planificar y ejecutar programas que propicien la participación de los integrantes de los diversos grupos parroquiales en todos los niveles de catequesis.
- Capacitar a los catequistas en una formación integral cristiana.
- Concientizar a la comunidad parroquial para una participación consciente en las celebraciones litúrgicas.
- Propiciar desde la Palabra y la Vida Teologal la transformación personal de cada cristiano manifestada a través de su testimonio logrando una comunidad viva.
- Revitalizar en la comunidad la vivencia de los sacramentos.
- Fomentar la integración de los miembros de los diversos grupos parroquiales a ejemplo de las primeras comunidades cristianas.
- Organizar la Pastoral Social Parroquial, a fin de atender a los más necesitados siguiendo el Carisma Vicentino.
- Animar y vivir la dimensión misionera de la fe en todas las instancias Pastorales, Catequesis, Sacramentos.
- Realizar el análisis situacional de la Parroquia a fin de obtener el diagnóstico respectivo para una posterior elaboración de los documentos de gestión parroquial.

### **1.1.5. Unidades Estratégicas de Negocios**

- Pastoral Catequética
- Pastoral Juvenil
- Pastoral Litúrgica

- Pastoral Social
- Pastoral de Misiones
- Pastoral Familiar
- Pastoral de Piedad Popular

## **1.2. Análisis externo**

### **1.2.1. Análisis del entorno general**

#### **A. Factores económicos**

La parroquia sustenta sus gastos, como son luz, agua, materiales de oficina y otros, mediante las limosnas que se pide a la feligresía y algunas actividades como las rifas.

En ocasiones se realizan actividades como rifas para poder implementar ciertos materiales que les son necesarios.

Podemos decir, en cierto modo, que la economía afecta indirectamente a la parroquia, puesto que si afecta a la feligresía como para verse forzada a tener que gastar menos, las personas que ofrecen limosnas se reducirían o reduciría el monto que estos pueden ofrecer. Al tener esta baja, la parroquia también se ve obligada a tener que recortar gastos o tener que prescindir de ciertas cosas.

#### **B. Factores tecnológicos**

Actualmente se encuentra involucrada a sus necesidades a hacer uso de equipos tecnológicos para sus actividades de oficina como son el uso de computadoras y señal wifi para conexión a internet, además de contar con una laptop y proyector para exposiciones de temas que se necesiten hacer a grandes grupos de personas.

Toman a la tecnología como una ayuda para sus actividades, más el avance tecnológico no afecta en gran parte al desarrollo de las mismas.

#### **C. Factores políticos**

La política en sí no afecta en gran escala al desarrollo de las actividades que realiza la parroquia. Al vivir en un país en la que el catolicismo es aceptado por la gran mayoría de personas y al ser su religión principal, se puede desenvolver de manera libre y segura.

En comparación de otros países donde la práctica de la religión católica es prohibida, y por ende el desarrollo de las parroquias que allí se encuentran es afectado enormemente por las políticas de estado que allí se rigen.

#### **D. Factores sociales**

Decir si la sociedad afecta o no a la parroquia es un poco difícil, puesto que sabemos que sin seguidores, una religión desaparece o pierde fuerza enormemente. No obstante, la parroquia tiene como función principal difundir las enseñanzas de Jesús, el cual tiene poder para cambiar a las personas. Mediante esta enseñanza es que la

parroquia posee tantas personas que apoyan a sus actividades, tanto económicas como misioneras.

La iglesia se preocupa por la sociedad donde se encuentra, pero no la preocupación de perder seguidores, sino por la preocupación de anunciar la palabra de Dios a todas las personas que desconocen de él.

#### **E. Factores demográficos**

La población pisqueña es, en su gran mayoría, de religión católica. Sin embargo hay presencia cada vez más fuerte de otros grupos religiosos, como son: mormones, evangelistas, adventistas, etc.

La presencia de dichos grupos pone en alerta a la parroquia, la cual se esfuerza por llevar evangelizar a las personas por medio de sus misioneros, en especial en los lugares alejados del centro de la provincia.

### **1.3. Análisis Interno**

#### **1.3.1. Recursos y capacidades**

##### **A. Recursos tangibles**

- 4 Computadoras
- 4 Impresoras
- 1 Router
- 1 Laptop
- 1 Proyector Multimedia
- Muebles de Oficina
- Sillas
- Bancas
- 2 Sagrarios
- Libros de registro de sacramentos
- Actas Pre-Preparadas
- 1 Camioneta
- 1 Capilla interna

##### **B. Recursos intangibles**

- Conocimiento de la Doctrina Cristiana
- Conocimiento Teológico
- Relación con la comunidad
- Valores morales

##### **C. Capacidades organizativas**

- Liderazgo
- Recurso humano en constante formación
- Adecuada administración financiera
- Comunicación horizontal
- Apoyo mutuo

#### **D. Análisis de recursos y capacidades**

De lo mencionado anteriormente se puede decir que la parroquia administra en forma correcta gran parte de sus recursos.

El párroco junto con el Consejo Parroquial, son quienes se esfuerzan en llevar a delante la parroquia, cumpliendo con sus actividades programadas y haciendo uso al máximo en que se pueda de los recursos disponibles.

Sin embargo, hay un punto que se ha descuidado y es la falta de documentos que apoyen la administración parroquial. Es cierto que todos en conjunto pueden complementar ideas y así repartir eficazmente sus recursos, pero en muchas ocasiones es necesario una base documental donde se especifique los acuerdos a los cuáles se han llegado y que todas las unidades deben de respetar.

##### **1.3.2. Análisis de la cadena de valor**

###### **A. Actividades primarias**

- Misas
- Bautismos
- Confirmaciones
- Matrimonios

###### **B. Actividades de apoyo**

- Catequesis
- Misiones

#### **1.4. Análisis Estratégico**

##### **1.4.1. Análisis FODA**

###### **A. Fortalezas**

- F1: Equipos organizados para catequesis.
- F2: Asesoramiento permanente
- F3: Formación permanente de catequistas
- F4: Equipo de liturgia permanente
- F5: Asistencia de las comunidades en la piedad popular
- F6: Horario establecido para los servicios sacramentales
- F7: Comunidad que ofrecen servicio social
- F8: Equipo misionero permanente
- F9: Creación de comunidades eclesiales de base
- F10: Grupos parroquiales que trabajan en la evangelización
- F11: Consejo Parroquial organizado
- F12: Personal administrativo y de servicio permanente y remunerado.
- F13: Ubicación apropiada
- F14: Ambientes apropiados

## **B. Oportunidades**

- O1: Acogida por parte de la comunidad de los programas de catequesis
- O2: Cursos para catequistas
- O3: Cursos de liturgia
- O4: Manifestación de la fe popular en fiestas
- O5: Maestros de religión capacitados
- O6: Apoyo de las instituciones locales y nacionales
- O7: Apoyo permanente de la comunidad en acciones de caridad
- O8: Acogida fraterna de la comunidad
- O9: Zonas que esperan el anuncio del Evangelio
- O10: Capillas en buenas condiciones

## **C. Debilidades**

- D1: Falta de compromiso de los catequistas
- D2: Falta de seguimiento a los catequizandos
- D3: Escaso número de catequistas
- D4: Escaso conocimiento de la Doctrina Social de la Iglesia
- D5: Falta de ambientes adecuados para las catequesis
- D6: Falta de integración entre los catequistas
- D7: Falta de seguimiento de los catequistas
- D8: Falta de coordinación entre el equipo de liturgia y el coro
- D9: Escasa formación litúrgica de los grupos parroquiales
- D10: Se cae en la rutina
- D11: Falta de cambio ante la realidad del evangelio
- D12: Escasez de vocaciones religiosas
- D13: Poco aprecio de los sacramentos por parte de los grupos parroquiales
- D14: Limitada difusión de la doctrina cristiana
- D15: Falta de pastoral social debidamente organizada
- D16: Pocos misioneros
- D17: Falta de jóvenes en la acción misionera
- D18: Falta de un Plan de Trabajo Misionero
- D19: No se cuenta con documentos de gestión parroquial

## **D. Amenazas**

- A1: Comunidad que considera los sacramentos como acto social
- A2: Familias con escasa formación cristiana
- A3: Escasa formación litúrgica de la comunidad
- A4: Propaganda persistente de otros grupos religiosos
- A5: Fe superficial
- A6: Falta de transmisión de la fe
- A7: Ambiente social escaso de valores cristianos y morales
- A8: Fe costumbrista hacia los sacramentos
- A9: Comunidad que confunde el sentido de caridad
- A10: Falta de seguridad

#### 1.4.2. Matriz FODA

Análisis Interno Análisis Externo	Fortalezas	Debilidades
Oportunidades	<p>E5: Profundizar la espiritualidad litúrgica</p> <p>E15: Colaborar con otras instituciones que realicen actividades de acción social</p> <p>E16: Ampliar y fortalecer el equipo misionero existente</p>	<p>E1: Concientizar a los coordinadores de grupo en el servicio de catequesis</p> <p>E2: Designar asesores para cada equipo de catequesis</p> <p>E4: Fortalecer el equipo de liturgia</p> <p>E10: Motivar la vocación sacerdotal</p> <p>E11: Vivir momentos gratos para vincularse y fraternizar</p> <p>E17: Organizar la Pastoral Juvenil</p>
Amenazas	<p>E6: Ofrecer reuniones de integración comunitaria</p> <p>E7: Talleres de formación misionera</p> <p>E12: Convocar a los grupos parroquiales que realizan acciones de caridad</p> <p>E14: Invocar el apoyo de instituciones y profesionales multidisciplinarios</p> <p>E19: Construir un Comité Pro-Implementación de todo el mobiliario y diversos equipos</p>	<p>E3: Realizar actividades de crecimiento espiritual</p> <p>E8: Renovar la pedagogía de preparación de los sacramentos</p> <p>E9: Promover la formación de padres y padrinos para que estos cumplan con su responsabilidad de educar en la fe a sus hijos y ahijados</p> <p>E13: Organizar el equipo encargado de la Pastoral Social</p> <p>E18: Análisis FODA</p>

Tabla 1 Matriz FODA

## **CAPÍTULO II: MARCO TEÓRICO DEL NEGOCIO Y DEL PROYECTO**

## **2.1. Marco teórico del Negocio**

### **2.1.1. Marco teórico del Negocio 1**

#### **Iglesia**

La iglesia de nuestros tiempos es la comunidad universal de los que creemos en Jesucristo. Transmite la fe por el Bautismo a los que entramos en ella, nos inserta en Cristo y nos acompaña para que podamos vivir y madurar como cristianos.<sup>1</sup>

#### **Sacramentos**

En la Iglesia tenemos signos o señales de la presencia salvadora de Jesús, instituidos por él mismo: son los sacramentos.

Los sacramentos son signos eficaces: transmiten realmente la gracia que anuncian.<sup>2</sup>

#### **Bautismo**

El bautismo es el sacramento común a todos los cristianos: es el primero pues nos inserta en la comunidad de los creyentes. La Iglesia lo administra según el mandato del Señor.<sup>3</sup>

#### **Confirmación**

El sacramento de la Confirmación completa, “confirma” los dones del Bautismo; en él recibimos la plenitud del Espíritu Santo, para vivir la madurez cristiana.

La Confirmación es necesaria “para recibir la gracia bautismal” (CEC, 1285). Quien opta por una vida hijo/a de Dios, pide este sacramento para recibir la fuerza del Espíritu Santo y ser testigo del amor de Dios. Se convierte en miembro pleno y responsable en la Iglesia.<sup>4</sup>

#### **Matrimonio**

En el sacramento del Matrimonio, quienes se aman se prometen fidelidad para toda la vida delante de Dios, dando libremente su consentimiento. Entonces su amor es transformado por el amor de Dios (CEC, 1639-1642).<sup>5</sup>

---

<sup>1</sup> Hna. Eleana S. (2013).Creo en la Santa Iglesia Católica. Catecismo de la Iglesia Joven (p. 78). Lima: Editorial Salesiana.

<sup>2</sup> Hna. Eleana S. (2013).Los Sacramentos de la Iglesia. Catecismo de la Iglesia Joven (p. 106). Lima: Editorial Salesiana.

<sup>3</sup> Hna. Eleana S. (2013).Sacramentos de Iniciación Cristiana. Catecismo de la Iglesia Joven (p. 110). Lima: Editorial Salesiana.

<sup>4</sup> Hna. Eleana S. (2013).Sacramentos de Iniciación Cristiana. Catecismo de la Iglesia Joven (p. 114). Lima: Editorial Salesiana.

<sup>5</sup> Hna. Eleana S. (2013).Sacramentos al Servicio de la Comunidad y la Misión. Catecismo de la Iglesia Joven (p. 136). Lima: Editorial Salesiana.

### **2.1.2. Marco teórico del Negocio 2**

#### **Partida de Bautismo**

Es un instrumento público y oficial por el cual se certifica solemnemente el hecho del bautismo con las circunstancias que definen la persona del bautizado.

Del hecho del bautismo brotan dos consecuencias canónicas importantísimas: la primera es el status de bautizado, de ser miembro de la Iglesia, sujeto de derechos canónicos y de obligaciones según su evolución humana; la partida de bautismo, pues, es la carta de nacionalidad espiritual de un sujeto. Esto es fundamental.<sup>6</sup>

## **2.2. Marco teórico del Proyecto**

### **2.2.1. Gestión del Proyecto**

#### **PMBOK**

Project Management Body of Knowledge o “Cuerpo de Conocimiento de la Gestión de Proyectos”. Es una norma reconocida en la profesión de la dirección de proyectos. Por norma se hace referencia a un documento formal que describe normas, métodos, procesos y prácticas establecidos. Al igual que en otras profesiones, como la abogacía, la medicina y las ciencias económicas, el conocimiento contenido en esta norma evolucionó a partir de las buenas prácticas reconocidas por profesionales dedicados a la dirección de proyectos, quienes contribuyeron a su desarrollo.<sup>7</sup>

#### **Proyecto**

Un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único. La naturaleza temporal de los proyectos indica un principio y un final definidos. El final se alcanza cuando se logran los objetivos del proyecto o cuando se termina el proyecto porque sus objetivos no se cumplirán o no pueden ser cumplidos, o cuando ya no existe la necesidad que dio origen al proyecto. Temporal no necesariamente significa de corta duración.<sup>8</sup>

#### **Dirección de Proyectos**

La dirección de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con los requisitos del mismo. Se logra mediante la aplicación e integración adecuadas de los 42 procesos de la dirección de proyectos, agrupados lógicamente, que conforman los 5 grupos de procesos. Estos 5 grupos de procesos son:

---

<sup>6</sup> Monseñor Dr. Fidel Tubino. La Partida de Bautismo y el Estado Civil de las personas. Citado el 06/12/2013. Recuperado de

<http://revistas.pucp.edu.pe/index.php/derechopucp/article/viewFile/6485/6555>

<sup>7</sup> Project Management Institute Inc. (2008). Introducción. Guía del PMBOK 4ta Edición (p. 10). Pennsylvania: Project Management Institute Inc.

<sup>8</sup> Project Management Institute Inc. (2008). Introducción. Guía del PMBOK 4ta Edición (p. 11). Pennsylvania: Project Management Institute Inc.

- Iniciación,
- Planificación,
- Ejecución,
- Seguimiento y Control, y
- Cierre.<sup>9</sup>

### **Grupo de Procesos de Iniciación**

El Grupo del Proceso de Iniciación está compuesto por aquellos procesos realizados para definir un nuevo proyecto o una nueva fase de un proyecto ya existente, mediante la obtención de la autorización para comenzar dicho proyecto o fase.

Las actividades en este grupo son:

1. Acta de Constitución del Proyecto
2. Lista de Interesados<sup>10</sup>

### **Grupo de Procesos de Planificación**

El Grupo del Proceso de Planificación está compuesto por aquellos procesos realizados para establecer el alcance total del esfuerzo, definir y refinar los objetivos, y desarrollar la línea de acción requerida para alcanzar dichos objetivos.

Las actividades en este grupo son:

1. Desarrollar el Plan para la Dirección del Proyecto
2. Recopilar Requisitos
3. Definir el Alcance
4. Crear la EDT (Estructura de Desglose de Trabajo)
5. Cronograma
  - a. Definir las actividades
  - b. Secuenciar las actividades
  - c. Estimar los recursos de las actividades
  - d. Estimar la duración de las actividades
  - e. Desarrollar el cronograma
6. Presupuesto
  - a. Estimar los costos
  - b. Determinar el presupuesto
7. Planificar la calidad
8. Desarrollar el Plan de Recursos Humanos
9. Planificar las comunicaciones
10. Planificar la Gestión de Riesgos
  - a. Identificar los riesgos
  - b. Realizar análisis cualitativo de riesgos
  - c. Realizar análisis cuantitativo de riesgos
  - d. Planificar la respuesta a los riesgos
11. Planificar las adquisiciones<sup>11</sup>

---

<sup>9</sup> Project Management Institute Inc. (2008). Introducción. Guía del PMBOK 4ta Edición (p. 12). Pennsylvania: Project Management Institute Inc.

<sup>10</sup> Project Management Institute Inc. (2008). Procesos de la Dirección de Proyectos para un Proyecto. Guía del PMBOK 4ta Edición (p. 47). Pennsylvania: Project Management Institute Inc.

### **Grupo de procesos de Ejecución**

El Grupo del Proceso de Ejecución está compuesto por aquellos procesos realizados para completar el trabajo definido en el plan para la dirección del proyecto a fin de cumplir con las especificaciones del mismo.

Las actividades en este grupo son:

1. Dirigir y gestionar la ejecución del proyecto
2. Realizar el aseguramiento de la calidad
3. Adquirir el equipo de proyecto
4. Desarrollar el equipo de proyecto
5. Dirigir el equipo de proyecto
6. Distribuir la información
7. Gestionar las expectativas de los interesados
8. Efectuar las adquisiciones<sup>12</sup>

### **Grupo de procesos de Seguimiento y Control**

El grupo del Proceso de Seguimiento y Control está compuesto por aquellos procesos requeridos para supervisar, analizar y regular el progreso y el desempeño del proyecto, para identificar áreas en las que el plan requiera cambios y para iniciar los cambios correspondientes.

Las actividades en este grupo son:

1. Dar seguimiento y controlar el trabajo del proyecto
2. Realizar el control integrado de cambios
3. Verificar el alcance
4. Controlar el alcance
5. Controlar el cronograma
6. Controlar costos
7. Realizar control de calidad
8. Informar el desempeño
9. Dar seguimiento y controlar los riesgos
10. Administrar las adquisiciones<sup>13</sup>

### **Grupo de procesos de Cierre**

El Grupo del Proceso del Cierre está compuesto por aquellos procesos realizados para finalizar todas las actividades a través de todos los grupos de procesos de la dirección de proyectos, a fin de completar formalmente el proyecto, una fase del mismo u otras obligaciones contractuales.

Las actividades en este grupo son:

1. Cerrar el proyecto o fase
2. Cerrar las adquisiciones<sup>14</sup>

---

<sup>11</sup> Project Management Institute Inc. (2008). Procesos de la Dirección de Proyectos para un Proyecto. Guía del PMBOK 4ta Edición (pp. 49-57). Pennsylvania: Project Management Institute Inc

<sup>12</sup> Project Management Institute Inc. (2008). Procesos de la Dirección de Proyectos para un Proyecto. Guía del PMBOK 4ta Edición (pp. 57-61). Pennsylvania: Project Management Institute Inc.

<sup>13</sup> Project Management Institute Inc. (2008). Procesos de la Dirección de Proyectos para un Proyecto. Guía del PMBOK 4ta Edición (pp. 61-65). Pennsylvania: Project Management Institute Inc.

## Correspondencia entre grupos de procesos y áreas de conocimiento de la dirección de proyectos

Knowledge Areas	Project Management Process Groups				
	Initiating Process Group	Planning Process Group	Executing Process Group	Monitoring & Controlling Process Group	Closing Process Group
<b>4. Project Integration Management</b>	4.1 Develop Project Charter	4.2 Develop Project Management Plan	4.3 Direct and Manage Project Execution	4.4 Monitor and Control Project Work 4.5 Perform Integrated Change Control	4.6 Close Project or Phase
<b>5. Project Scope Management</b>		5.1 Collect Requirements 5.2 Define Scope 5.3 Create WBS		5.4 Verify Scope 5.5 Control Scope	
<b>6. Project Time Management</b>		6.1 Define Activities 6.2 Sequence Activities 6.3 Estimate Activity Resources 6.4 Estimate Activity Durations 6.5 Develop Schedule		6.6 Control Schedule	
<b>7. Project Cost Management</b>		7.1 Estimate Costs 7.2 Determine Budget		7.3 Control Costs	
<b>8. Project Quality Management</b>		8.1 Plan Quality	8.2 Perform Quality Assurance	8.3 Perform Quality Control	
<b>9. Project Human Resource Management</b>		9.1 Develop Human Resource Plan	9.2 Acquire Project Team 9.3 Develop Project Team 9.4 Manage Project Team		
<b>10. Project Communications Management</b>	10.1 Identify Stakeholders	10.2 Plan Communications	10.3 Distribute Information 10.4 Manage Stakeholder Expectations	10.5 Report Performance	
<b>11. Project Risk Management</b>		11.1 Plan Risk Management 11.2 Identify Risks 11.3 Perform Qualitative Risk Analysis 11.4 Perform Quantitative Risk Analysis 11.5 Plan Risk Responses		11.6 Monitor and Control Risks	
<b>12. Project Procurement Management</b>		12.1 Plan Procurements	12.2 Conduct Procurements	12.3 Administer Procurements 12.4 Close Procurements	

**Ilustración 1 Cuadro de Grupos de Procesos y Área de Conocimiento**

---

<sup>14</sup> Project Management Institute Inc. (2008). Procesos de la Dirección de Proyectos para un Proyecto. Guía del PMBOK 4ta Edición (pp. 65-66). Pennsylvania: Project Management Institute Inc.

## 2.2.2. Ingeniería del Proyecto

### RUP

El Proceso Unificado de IBM Rational (RUP) es un marco de proceso de ingeniería de software. Proporciona las mejores prácticas y orientación para el desarrollo de software de éxito y un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga las necesidades de sus usuarios dentro de un horario predecible y presupuestado.

El ciclo de vida se divide en cuatro fases: concepción, elaboración, construcción y transición. Cada fase se divide en una o más iteraciones. Por ejemplo inicio tiene una iteración, elaboración tiene dos iteraciones, la construcción tiene n iteraciones, y la transición tiene dos iteraciones. El número correcto de iteraciones por fase varía de proyecto a proyecto.<sup>15</sup>

### Arquitectura General de RUP

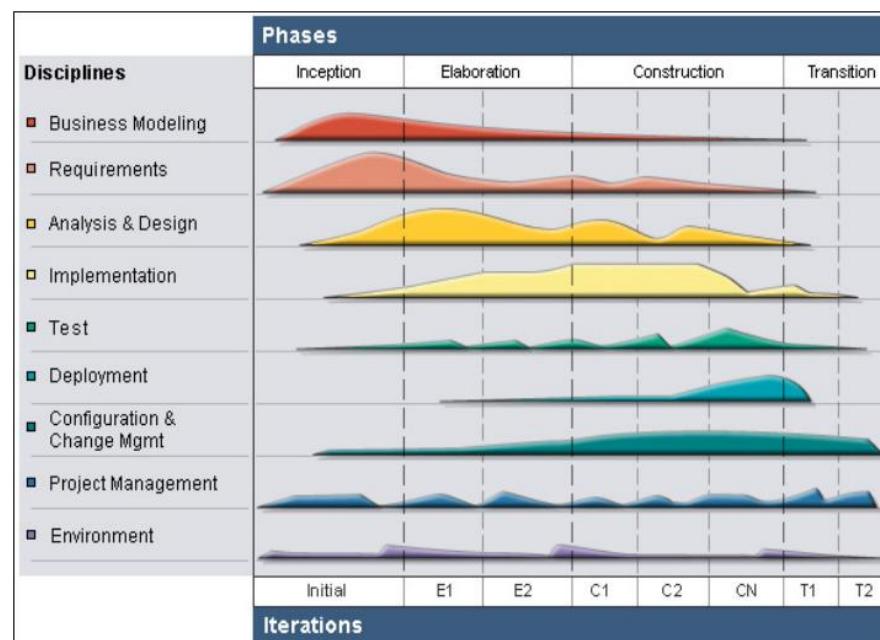


Ilustración 2 Arquitectura General de RUP

### Fase de Incepción

El objetivo primordial de la fase inicial es lograr la concurrencia entre todas las partes interesadas sobre los objetivos del ciclo de vida del proyecto. La fase de inicio es de importancia sobre todo para los nuevos esfuerzos de desarrollo, en los que existen riesgos de negocio y requisitos importantes que deben abordarse antes de que el proyecto pueda continuar. En los proyectos que se centraron en mejoras a un

<sup>15</sup> Cécile P, Mike E, Angelo F, Enrico M, Kathy C. (2007). Introduction to the IBM Rational Unified Process and its extension to Service-Oriented Architecture. The IBM Rational Unified Process for System Z (p. 12). New York: IBM Corporation.

sistema existente, la fase de inicio es más corta, pero todavía se centra en asegurar que el proyecto valga la pena hacer y se puede hacer.

Las actividades esenciales en esta fase son:

- Formular el alcance del proyecto
- Planificar y preparar el modelo de negocio
- Sintetizar la posible arquitectura
- Preparación del entorno para el proyecto<sup>16</sup>

### **Fase de Elaboración**

El objetivo de la fase de elaboración es proporcionar a la línea base de la arquitectura del sistema una base estable para la mayor parte del diseño y el esfuerzo de implementación de la Fase de Construcción. La arquitectura se desarrolla a partir de una consideración de los requisitos más importantes (las que tienen un gran impacto en la arquitectura del sistema) y una evaluación del riesgo. La estabilidad de la arquitectura se evalúa a través de uno o más prototipos arquitectónicos.

Las actividades esenciales en esta fase son:

- Definir, validar y proporcionar la línea de base de la arquitectura
- Redefinir la visión
- Crear y proporcionar la línea base de los planes de iteración detallados para la Fase de Construcción
- Redefinir el proceso de desarrollo y puesta en marcha del entorno de desarrollo
- Redefinir la arquitectura y seleccionar los componentes.<sup>17</sup>

### **Fase de Construcción**

El objetivo de la fase de construcción está clarificando los requisitos restantes y completar el desarrollo del sistema basado en la arquitectura de la línea de base. La fase de construcción es, en cierto sentido, un proceso de fabricación, donde se hace hincapié en la gestión de los recursos y el control de las operaciones para optimizar los costos, horarios y calidad. En este sentido, el modo de pensar de gestión experimenta una transición desde el desarrollo de la propiedad intelectual durante la creación y elaboración, para el desarrollo de productos de despliegue durante la construcción y la transición.

Las actividades esenciales en esta fase son:

- La administración de recursos, control y optimización de procesos
- Desarrollo de componentes completa y las pruebas en contra de los criterios de evaluación definidos

---

<sup>16</sup> Cécile P, Mike E, Angelo F, Enrico M, Kathy C. (2007). Introduction to the IBM Rational Unified Process and its extension to Service-Oriented Architecture. The IBM Rational Unified Process for System Z (pp. 18-19). New York: IBM Corporation.

<sup>17</sup> Cécile P, Mike E, Angelo F, Enrico M, Kathy C. (2007). Introduction to the IBM Rational Unified Process and its extension to Service-Oriented Architecture. The IBM Rational Unified Process for System Z (p. 21). New York: IBM Corporation.

- Determinación de las emisiones de productos según los criterios de aceptación para la visión<sup>18</sup>

### **Fase de Transición**

El foco de la fase de transición es asegurar que el software está disponible para sus usuarios. La fase de transición puede abarcar varias iteraciones e incluye las pruebas del producto en preparación para el lanzamiento y hacer ajustes menores basados en la retroalimentación de los usuarios. En este punto del ciclo de vida, respuesta de los usuarios ha de centrarse principalmente en afinar el producto, configuración, instalación y problemas de usabilidad, los principales problemas estructurales que deberían haber sido resueltos mucho antes en el ciclo de vida del proyecto.

Las actividades esenciales en esta fase son:

- La ejecución de los planes de despliegue
- Finalización de material de apoyo a los usuarios
- Probar el producto liberado en el sitio de desarrollo
- Creación de un lanzamiento de producto
- Obtener retroalimentación de los usuarios
- Afinar el producto basado en la retroalimentación
- Hacer que el producto esté disponible para los usuarios<sup>19</sup>

### **N-Capas**

Las capas son agrupaciones horizontales lógicas de componentes de software que forman la aplicación o el servicio. Nos ayudan a diferenciar entre los diferentes tipos de tareas a ser realizadas por los componentes, ofreciendo un diseño que maximiza la reutilización y, especialmente, la mantenibilidad.

El dividir una aplicación en capas separadas que desempeñan diferentes roles y funcionalidades, nos ayuda a mejorar el mantenimiento del código; nos permite también diferentes tipos de despliegue y, sobre todo, nos proporciona una clara delimitación y situación de dónde debe estar cada tipo de componente funcional e incluso cada tipo de tecnología.

La clave de una aplicación en N-Capas está en la gestión de dependencias. En una arquitectura N-Capas tradicional, los componentes de una capa pueden interactuar solo con componentes de la misma capa o bien con otros componentes de capas inferiores. Esto

---

<sup>18</sup> Cécile P, Mike E, Angelo F, Enrico M, Kathy C. (2007). Introduction to the IBM Rational Unified Process and its extension to Service-Oriented Architecture. The IBM Rational Unified Process for System Z (p. 23). New York: IBM Corporation.

<sup>19</sup> Cécile P, Mike E, Angelo F, Enrico M, Kathy C. (2007). Introduction to the IBM Rational Unified Process and its extension to Service-Oriented Architecture. The IBM Rational Unified Process for System Z (pp. 24-26). New York: IBM Corporation.

ayuda a reducir las dependencias entre componentes de diferentes niveles.<sup>20</sup>

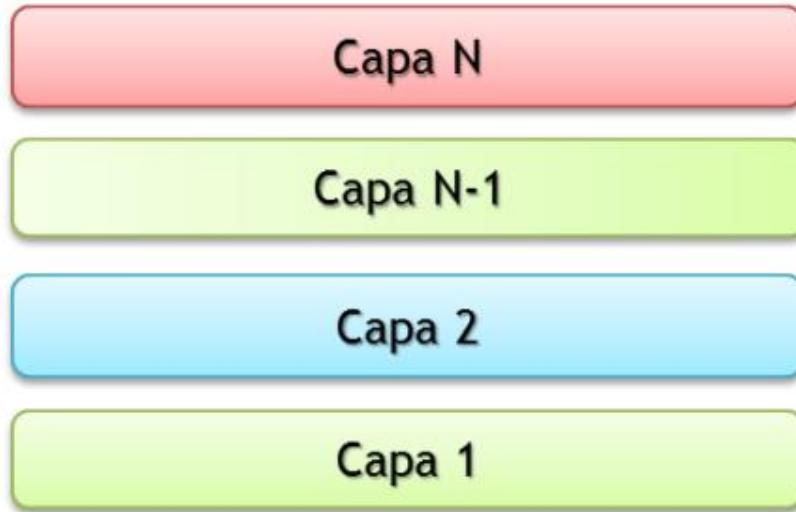


Ilustración 3 Diseño Básico de Capas

### **Beneficios de uso de Capas**

- El mantenimiento de mejoras en una solución será mucho más fácil porque las funciones están localizadas. Además las capas deben estar débilmente acopladas entre ellas y con alta cohesión internamente, lo cual posibilita variar de una forma sencilla diferentes implementaciones/combinaciones de capas.
- Otras soluciones deberían poder reutilizar funcionalidad expuesta por las diferentes capas, especialmente si se han diseñado para ello.
- Los desarrollos distribuidos son mucho más sencillos de implementar si el trabajo se ha distribuido previamente en diferentes capas lógicas.
- La distribución de capas (layers) en diferentes niveles físicos (tiers) puede, en algunos casos, mejorar la escalabilidad. Aunque este punto hay que evaluarlo con cuidado, pues puede impactar negativamente en el rendimiento.<sup>21</sup>

<sup>20</sup> De la Torre Llorente C, Zorrilla Castro U, Calvarro Nelson J, Ramos Barroso M. (2010). Arquitectura Marco N-Capas. Guía de Arquitectura N-Capas Orientada al Dominio con .NET 4.0 (pp. 23-24). España: Krasis Press.

<sup>21</sup> De la Torre Llorente C, Zorrilla Castro U, Calvarro Nelson J, Ramos Barroso M. (2010). Arquitectura Marco N-Capas. Guía de Arquitectura N-Capas Orientada al Dominio con .NET 4.0 (p. 26). España: Krasis Press.

### **2.2.3. Soporte del Proyecto**

#### **Valor Ganado**

El Valor Ganado (EV) es una función del tiempo y representa el avance de un proyecto. Se mide en unidades de valor económico (\$). Matemáticamente, el dominio del Valor Ganado (EV) es tiempo y el rango es valor económico (\$). El Valor Ganado (EV) está claramente asociado con el Valor Planeado (PV) porque el valor que EV tiene, cuando se supera un hito (o se alcanza un objetivo), es el mismo que tiene PV en las mismas circunstancias. En otras palabras, cuando es ejecutada una unidad de trabajo, EV adiciona el mismo valor que fue planeado para esa unidad de trabajo en la curva PV, no importa cuánto fue el costo real de terminar esa unidad de trabajo.<sup>22</sup>

#### **Métrica de Calidad**

Una métrica de calidad es una definición operativa que describe, en términos muy específicos, un atributo del producto o del proyecto, y la manera en que el proceso de control de calidad lo medirá. Una medición es un valor real. La tolerancia define la variación permisible de las métricas. Por ejemplo, una métrica relacionada con el objetivo de calidad de mantenerse dentro del límite de  $\pm 10$  por ciento del presupuesto aprobado puede consistir en medir el costo de cada entregable y determinar el porcentaje de desviación con respecto al presupuesto aprobado para ese entregable. Las métricas de calidad se emplean en los procesos de aseguramiento de la calidad y de control de calidad.<sup>23</sup>

#### **Evaluaciones del Desempeño del Equipo**

El equipo de dirección del proyecto realiza permanentemente evaluaciones formales o informales del desempeño del equipo del proyecto. Al evaluar continuamente el desempeño del equipo del proyecto, pueden llevarse a cabo acciones para resolver los problemas, modificar la comunicación, abordar los conflictos y mejorar la interacción del equipo.<sup>24</sup>

#### **Gestión de la Configuración**

En este flujo de trabajo se describe la forma de controlar los numerosos artefactos producidos por las muchas personas que trabajan en un proyecto común. El control ayuda a evitar confusión costosa, y asegura que los artefactos resultantes no están en conflicto debido a algunos de los siguientes tipos de problemas:

---

<sup>22</sup> Project Management Office S.A. Gestión del Valor Ganado. Citado el 06/12/2013. Recuperado de <http://www.projectmanagementofficesa.com/biblioteca/costo/evm.asp>

<sup>23</sup> Project Management Institute Inc. (2008). Gestión de la Calidad del Proyecto. Guía del PMBOK 4ta Edición (p. 175). Pennsylvania: Project Management Institute Inc.

<sup>24</sup> Project Management Institute Inc. (2008). Gestión de los Recursos Humanos del Proyecto. Guía del PMBOK 4ta Edición (p. 207). Pennsylvania: Project Management Institute Inc.

- Actualización Simultánea.- Cuando dos o más trabajadores trabajan por separado en el mismo artefacto, el último en hacer cambios destruye el trabajo de los primeros.
- Notificación Limitada.- Cuando un problema se resuelve en artefactos compartidos por varios desarrolladores, y algunos de ellos no son notificados del cambio.
- Múltiples versiones.- Más programas grandes se desarrollan en las versiones evolutivas. Una liberación podría estar en uso de los clientes, mientras que otro está en la prueba, y el tercero todavía está en desarrollo. Si se encuentran problemas en cualquiera de las versiones, las correcciones tienen que propagarse entre ellos. La confusión que puede surgir conduce a soluciones costosas y re-trabajo, a menos que los cambios sean cuidadosamente controlados y supervisados.<sup>25</sup>

---

<sup>25</sup> IBM. (2005).Core Workflows. *Rational Unified Process Best Practices for Software Development Teams* (p. 13). Lexington: Rational Software White Paper

## **CAPÍTULO III: INICIO Y PLANIFICACIÓN DEL PROYECTO**

### **3.1. Gestión del proyecto**

#### **3.1.1. Iniciación**

##### **A. Nacimiento del proyecto**

El presente proyecto nace de la necesidad que tiene la parroquia San Clemente de Pisco de una herramienta más eficiente para desarrollar sus actividades con respecto al control de sacramentos y asignación de personal.

##### **B. Justificación**

El proceso de registro de sacramentos del feligrés y su búsqueda para la impresión de las actas se hace de manera totalmente manual, lo cual ocasiona que un feligrés deba esperar como mínimo un día entero para obtener el acta del sacramento, generando costos de transporte y tiempo adicionales.

De la misma manera, la asignación del personal a las distintas comunidades de la parroquia no se actualiza como se debe y en muchos casos se desconoce con que miembros cuenta determina comunidad, lo que ocasiona que no se puedan tomar adecuadas decisiones al momento de realizar misiones y otras actividades que concierne a las comunidades.

##### **C. Importancia**

Es de vital importancia para la parroquia San Clemente pues contarán con una herramienta más eficiente y eficaz que el simple proceso manual actual. La implementación de un sistema software ahorraría tiempo de búsqueda de las actas y el feligrés no tendría que realizar viajes extras ni realizar gastos de transporte adicionales, lo que llevaría a una mayor satisfacción del servicio.

Así mismo, se contaría con una herramienta, que no se tiene actualmente, para actualizar y conocer que personal forma parte de una determinada comunidad dentro de la parroquia.

#### D. Acta de constitución del proyecto

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	JC	LL	VD	03/05/2013	Original
2.0	JC	LL	VD	30/09/2013	Actualización

#### ACTA DE CONSTITUCIÓN DEL PROYECTO

<b>Nombre del proyecto:</b>	Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal	<b>Cód. Proyecto:</b>	Proy-001
<b>Patrocinador del proyecto:</b>	Padre Vicente Diez Varona / Párroco		
<b>Gerente del proyecto:</b>	Juan Guillermo Castillo Hernández		
<b>Cliente del Proyecto:</b>	Parroquia San Clemente		
<b>Preparado por:</b>	Juan Guillermo Castillo Hernández		
<b>Situación actual:</b>	<p>La parroquia San Clemente para brindar sus diversos servicios de atención a la feligresía católica cristiana del cercado de Pisco cuenta con una estructura orgánico-funcional conformado por los siguientes estamentos: Oficina Parroquial, Secretaría y Grupos Parroquiales. En el área de secretaría, se realiza el registro de sacramentos recibidos por la feligresía y expedición de sus actas correspondientes, proceso que en la actualidad se viene desarrollando de manera manual, es decir, con la utilización de cuadernos o libros que son llenados a manuscrito, este proceso trae como consecuencia registros inadecuados en su escritura y excesivo tiempo para la entrega de los mismos.</p> <p>A la par de esta situación, la Parroquia no posee un adecuado control del personal asignado a las comunidades parroquiales. Se desconoce</p>		

con exactitud la cantidad de catequistas y misioneros asignados a las comunidades. Todos los informes son en su mayoría son verbales por parte de los responsables.

Por lo antes mencionado, la parroquia presenta un escenario no favorable para la atención de sus usuarios a quienes genera en muchos de los casos un grado de insatisfacción alta; así mismo, una falta de control objetiva y real de los ingresos que debe percibir dicha entidad religiosa. También se puede observar que presenta dificultades al no conocer el número de catequistas/misioneros asignados a guiar a las comunidades, lo cual origina que no se tomen medidas adecuadas para su crecimiento respectivo.

#### **Propósito / Necesidad de negocio:**

**La Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal** permitirá corregir el escenario actual generando un impacto positivo dentro de la estructura orgánica-funcional de la parroquia haciendo que sus procesos administrativos sean ágiles, eficaces y eficientes; con relación a los usuarios, disminuirá su grado de insatisfacción haciendo la entrega de documentos en periodos más cortos, y fáciles de ubicación, control y expedición.

Por otro lado, permitirá a la parroquia saber el número de personal con el que cuenta, condiciones en que se encuentra lo cual permitirá al párroco una distribución más eficiente y acertada de su personal de formación a las distintas comunidades parroquiales, permitiendo que estas crezcan y se desarrolleen en la proporción esperada.

#### **Descripción de proyecto**

Se implementará un sistema de escritorio, el cual optimizará el control del registro de los sacramentos recibidos por la feligresía y su respectiva expedición de actas además de analizar la información a través de reportes continuos, considerando el registro de los datos de crítica importancia para la parroquia, también brindará innovación al incorporar la asignación del personal parroquial. Este proyecto se desarrollará teniendo en cuenta las buenas prácticas del PMI, utilizando la guía PMBOK en su 4<sup>ta</sup> edición.

#### **Descripción del producto y entregables:**

#### **Entregables del producto**

- Sub-Sistema de Control de Sacramentos
  - Módulo de Feligrés

<ul style="list-style-type: none"> <li>○ Módulo de Sacramentos</li> <li>○ Módulo de Búsqueda y Reportes</li> <li>● Sub-Sistema de Asignación de Personal           <ul style="list-style-type: none"> <li>○ Módulo de Personal</li> <li>○ Módulo de Comunidades</li> <li>○ Módulo de Búsqueda y Reportes               <ul style="list-style-type: none"> <li>▪ Personal asignado a comunidades</li> </ul> </li> </ul> </li> <li>● Módulo de BackUp's</li> </ul> <p>Documentación</p> <ul style="list-style-type: none"> <li>● Manual técnico</li> <li>● Manual usuario</li> </ul> <p><b>Entregables del Py</b></p> <ul style="list-style-type: none"> <li>● Acta de constitución del Proyecto</li> <li>● Cronograma de Planificación y Ejecución del Proyecto</li> <li>● Presupuesto General del Proyecto</li> </ul>
<b>Criterios de aceptación:</b>
Compatibilidad con el sistema operativo Windows XP profesional y Windows 7 starter.  Lenguaje a desarrollar C# con base de datos MySql.  Utilizar la metodología RUP para el desarrollo.  Entorno amigable: interfaz gráfica de usuario, utilizando un conjunto de imágenes y objetos gráficos y acciones disponibles en la interfaz.
<b>Supuestos, restricciones:</b>
Que la plataforma tecnológica se encuentre en óptimas condiciones para llevar a cabo la correcta implantación del sistema.  El costo del proyecto no debe sobre pasar del monto base fijado de S/. 13,065.80.  El plazo del desarrollo del proyecto debe estar dentro de los 08 meses planteados para la entrega del Proyecto.  Al siguiente día de cumplido el plazo de desarrollo el Proyecto, este ya debe encontrarse instalado y ejecutándose correctamente dentro de la organización.
<b>Riesgos Iniciales:</b>
Cambio en la normatividad o políticas dentro de la organización.

<p>Infraestructura tecnológica inadecuada.</p> <p>Incompatibilidad entre las herramientas utilizadas para el desarrollo del Proyecto.</p> <p>Falta de colaboración por parte de los integrantes de la organización.</p> <p>No existencia de documentación solicitada.</p> <p>Falta de conocimiento en tecnología por parte del personal que usara el software.</p>		
Objetivos de Proyecto	Criterios de éxito	Persona que aprueba
<b>Alcance</b>		
<p>Se entregaran:</p> <p>Sub-Sistema de Control de Sacramentos</p> <ul style="list-style-type: none"> <li>• Módulo de Feligrés</li> <li>• Módulo de Sacramentos</li> <li>• Módulo de Búsqueda y Reportes</li> </ul> <p>Sub-Sistema de Asignación de Personal</p> <ul style="list-style-type: none"> <li>• Módulo de Personal</li> <li>• Módulo de Comunidades</li> <li>• Módulo de Búsqueda y Reportes</li> </ul> <p>Módulo de Backup's</p>	<p>Priorizando la entrega del primer subsistema: Sistema de Control de Sacramentos en la primera parte del desarrollo del Proyecto, este aplicativo se entregara con el funcionamiento correcto de todos los modulo planteados.</p> <p>Así mismo, la segunda entrega del siguiente subsistema: Sistema de Asignación de Personal es obligatoria con todos sus módulos funcionando correctamente.</p> <p>El Sistema de Control de Sacramentos debe ser entregado el 11/09/2013 y el Sistema de Asignación de Personal debe ser entregado el 16/10/2013.</p>	Vicente Diez / Párroco
<b>Tiempo</b>		
El proyecto finalizará el 21 de Noviembre del 2013	No se debe exceder de la fecha fijada en el presente documento.	Vicente Diez / Párroco
<b>Costo</b>		

El costo del Proyecto está fijado en S/. 11,878.00 más una reserva de S/. 1,187.80, haciendo un total de S/. 13,065.80.	El Proyecto finalice dentro del costo establecido.	Juan Castillo / Gerente del Proyecto
<b>Calidad</b>		
Se aplicará el ciclo de vida de desarrollo de software (SDLC).  Se evaluará la calidad del Producto de Software validando con el cliente las funciones del sistema, con los requerimientos que este debe cumplir.	Gestión y Seguimiento del Proyecto con un control estricto del cumplimiento de calidad de los Entregables.  Cumplir con lo indicado en los estándares de calidad definidos para el proyecto.  Instalación y ejecución correcta la empresa del producto.	Luis Loo Parian / Asesor
<b>Otros</b>		
<b>Resumen de Hitos:</b>		<b>Fecha esperado</b>
Entrega del primer subsistema: Sistema de Control de Sacramentos		11/09/2013
Entrega del Segundo Subsistema: Sistema de Asignación de Personal		16/10/2013
Pruebas Integrales: Sistema de Control de Sacramentos y Asignación de Personal		11/11/2013
Capacitación: Sistema de Control de Sacramentos y Asignación de Personal		19/11/2013
Cierre del Contrato		21/11/2013
<b>Presupuesto estimado</b>		
El presupuesto estimado para este proyecto es de S/. 13,065.80, el cual será asumido por el Gerente de Py.		
<b>Autoridad del gerente de proyecto</b>		
<b>Personal del proyecto</b>		
Se encargara de solicitar el personal idóneo para el proyecto.		

Establecerá los horarios de trabajo del equipo de proyecto, de acuerdo al avance del mismo.	
<b>Decisiones acerca de la variación del presupuesto.</b>	
Evaluar el seguimiento de nuevos requerimientos y remitir dichos resultados al sponsor para su aprobación.	
<b>Decisiones técnicas</b>	
Determina la metodología a utilizar en el desarrollo del Proyecto. Determina el ciclo de vida a utilizar en el desarrollo del Proyecto. Establecer las herramientas a utilizar en el desarrollo del producto.	
<b>Resolución de conflictos</b>	
Se limitará a resolver los conflictos que se presenten dentro del equipo de proyecto, como: falta de comunicación, diferencias existentes entre los integrantes del Proyecto.	
<b>Escalabilidad por limitación de autoridad del gerente de proyecto</b>	
Cualquier problema que se presente durante el desarrollo del proyecto y no esté contemplado dentro de la normativa o documentación vigente, será resuelto en primera instancia por el comité ejecutivo del Proyecto en coordinación con el personal responsable del área administrativa dentro de las cuales se presenten los conflictos. Y en segunda instancia el párroco con el responsable del proyecto quienes tomarán las medidas necesarias para resolver el conflicto.	
<b>Otros</b>	
<b>Aprobaciones</b>	
 _____ Juan Guillermo Castillo Hernández Gerente del Proyecto 04/10/2013	 _____ Vicente Diez Varona Párroco Patrocinador del Py 04/10/2013

## **E. Identificación de los interesados**

- Vicente Diez Varona => Párroco
- Luisa Cajo => Secretaria
- Juan Castillo => Gerente de Proyecto
- Luis Loo => Asesor
- Hugo Núñez => Director de Escuela

### **3.1.2. Planificación**

#### **A. Alcance**

##### **Definición del alcance del proyecto y alcance del producto**

###### Alcance del proyecto

Desarrollar un sistema software para los procesos de Control de Sacramentos y Asignación de Personal.

###### Alcance del producto

Desarrollar un Sistema de control de sacramentos y asignación de personal que permita:

- Registrar datos específicos del feligrés.
- Registrar datos específicos de cada sacramento.
- Permitir la impresión de actas de sacramentos.
- Registrar datos específicos del personal.
- Registrar datos específicos de las comunidades.
- Llevar el control de la asignación de personal a las diferentes comunidades.
- Permitir la impresión de reportes de asignación.

## Requerimientos

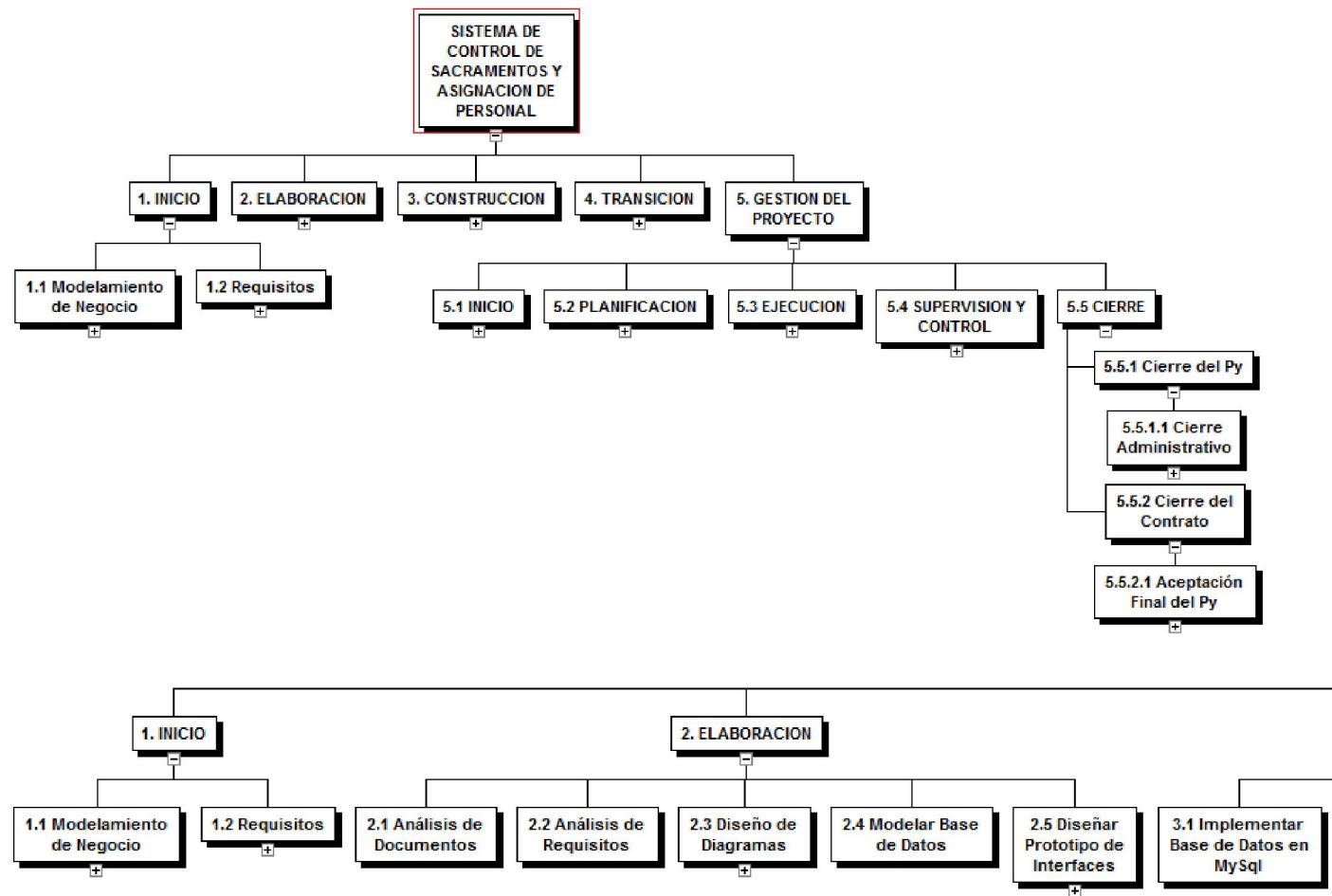
Involucrado	Cod. Req	Descripción del requerimiento	Categoría	Prioridad	Criterio de aceptación
Secretaria	Req-001	Registrar los datos específicos del Bautismo: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Natural de, Nombres de la Madre, Natural de, Lugar de nacimiento, Fecha de nacimiento, Fecha de Bautismo, Nombre del Padrino, Nombre de la Madrina, Ministro o Celebrante, Nota Marginal.	Req. Funcional	Alta	Cero errores al momento de llevar a cabo el registro del bautismo, durante las pruebas de Control de Calidad
Secretaria	Req-002	Registrar los datos específicos de la Confirmación: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Nombres de la Madre, Lugar de nacimiento, Fecha de nacimiento, Lugar de Bautismo, Fecha de Bautismo, Nº Libro – Fojas – Número del Bautismo, Fecha de la Confirmación, Nombre del Padrino o Madrina, Ministro o Celebrante, Anotaciones.	Req. Funcional	Alta	Cero errores al momento de llevar a cabo el registro de la confirmación, durante las pruebas de Control de Calidad
Secretaria	Req-003	Registrar los datos específicos del Matrimonio: Nombre de la Parroquia, Nº Libro – Fojas – Número, Nombres del Esposo, Nacido en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Nombres de la Esposa, Nacida en, Edad, Nombres	Req. Funcional	Alta	Cero errores al momento de llevar a cabo el registro del matrimonio, durante las pruebas de Control de Calidad.

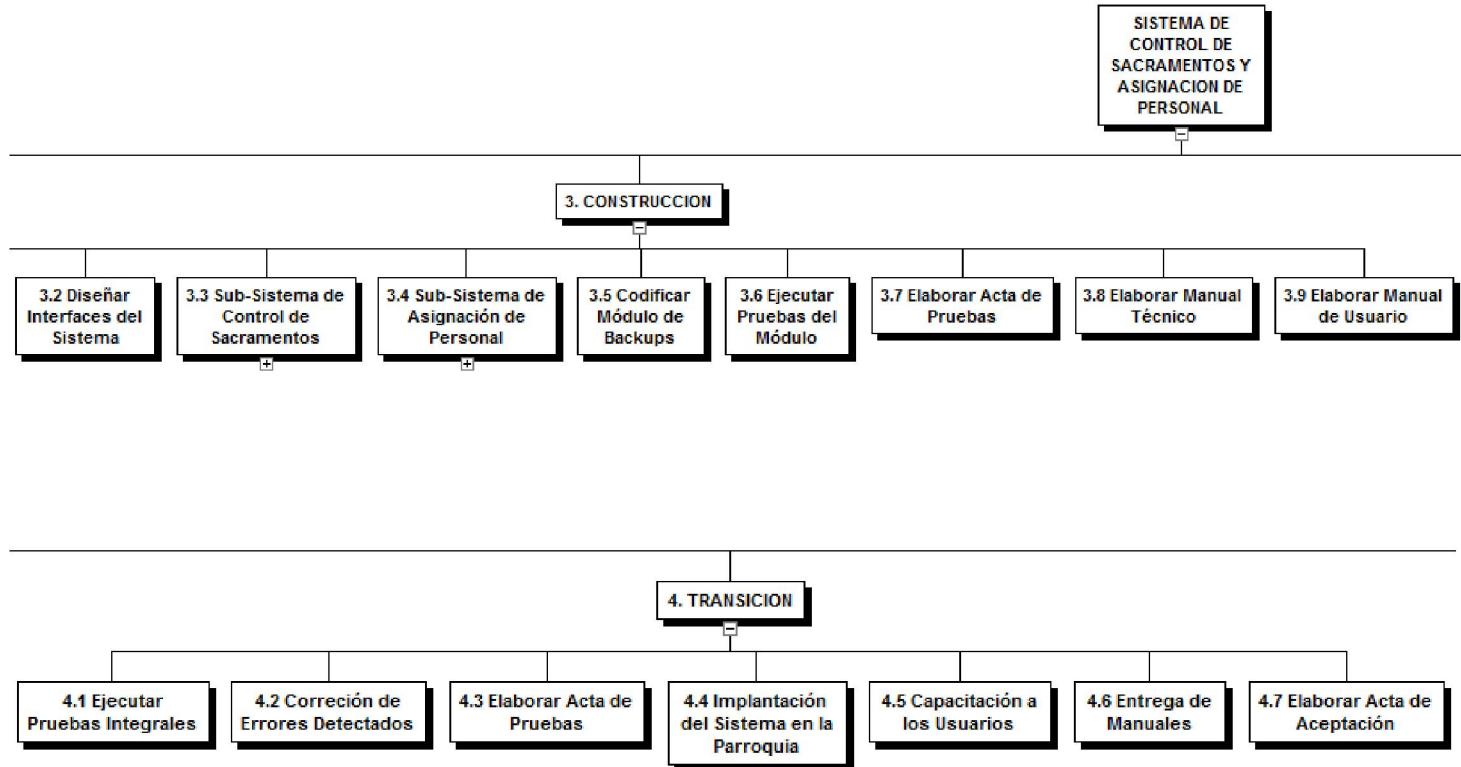
		del Padre, Nombres de la Madre, Lugar de Bautismo, Fecha de Bautismo, Fecha del Matrimonio, Nombre del Padrino, Nombre de la Madrina, Nombre de los Testigos (2), Ministro o Celebrante, Anotaciones.			
Secretaria	Req-004	No permitir la modificación de los registros de sacramentos para los datos: feligrés y libro, folio, número del libro	Req. Funcional	Alta	Inhabilitar dichos datos al momento de actualizar algún sacramento
Secretaria	Req-005	Ordenar los registros del feligrés alfabéticamente por apellidos.	Req. Funcional	Alta	Ordenar por orden alfabético de apellidos al realizar la búsqueda de registros.
Secretaria	Req-006	Buscar los sacramentos por el Apellido Paterno del feligrés.	Req. Funcional	Alta	Filtrado correcto de los registros por el Apellido Paterno del Feligrés.
Párroco	Req-007	Registrar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo.	Req. Funcional	Alta	Cero errores al momento de llevar a cabo el registro del personal, durante las pruebas de Control de Calidad.
Párroco	Req-008	Registrar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo.	Req. Funcional	Alta	Cero errores al momento de llevar a cabo el registro de las comunidades, durante las pruebas de Control de Calidad.
Secretaria	Req-009	Obtener una búsqueda rápida de partidas para dar un mejor servicio a los feligreses.	Req. No Funcional	Alta	Filtrado de datos correctamente y ordenado alfabéticamente.

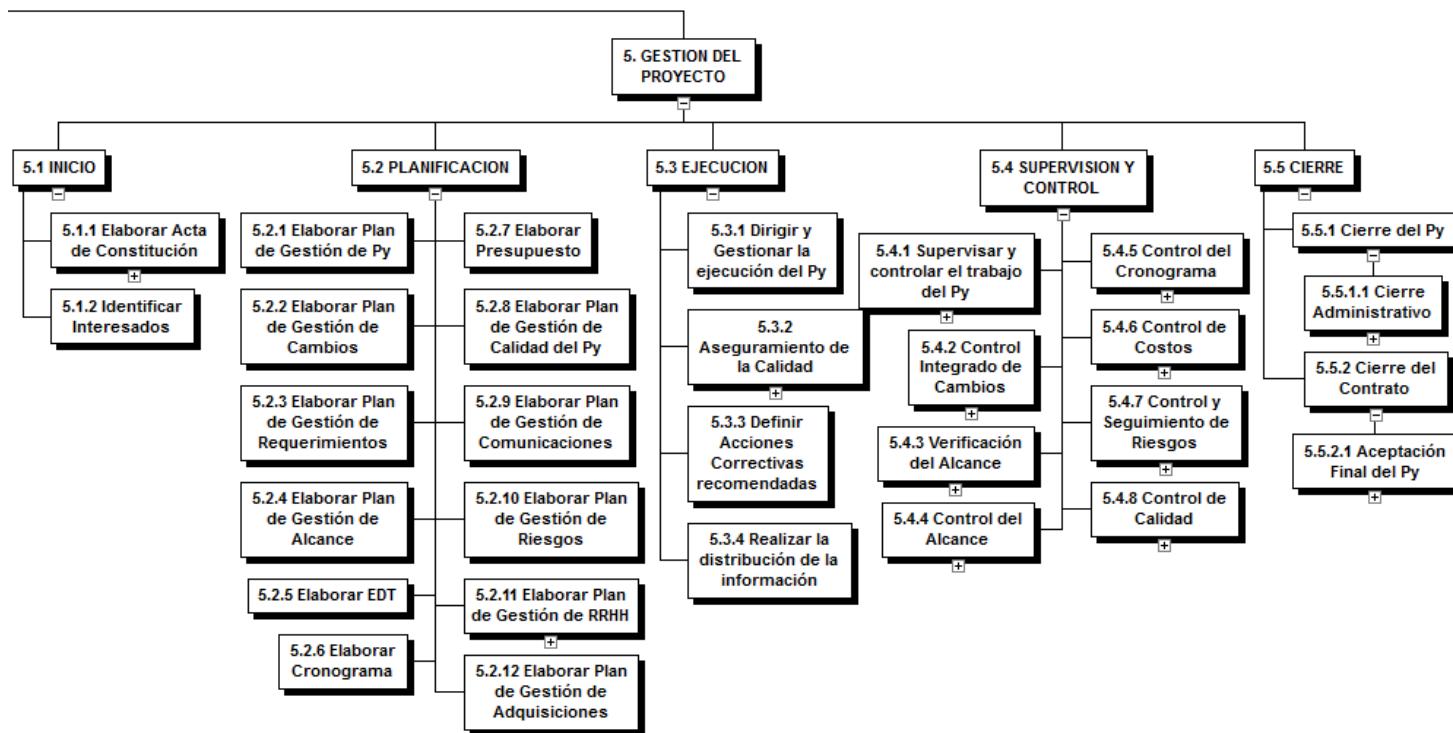
Párroco	Req-010	Obtener un Sistema Software con interfaz amigable y fácil de usar	Req. No Funcional	Media	Interfaz adecuada a los requisitos
Párroco	Req-011	Obtener un Sistema confiable y preciso.	Req. No Funcional	Media	Cero errores al momento de realizar las pruebas generales de Control de Calidad.
Secretaria	Req-012	Realizar copias de seguridad de los registros.	Req. No Funcional	Alta	Función correcta del módulo de BackUp's.
Párroco	Req-013	El nuevo sistema debe desarrollarse dentro del plazo establecido.	Req. No Funcional	Muy Alta	Cumplimiento del cronograma

Tabla 2 Requerimientos del Proyecto

## EDT







## **Entregables**

### **Principales Entregables del Producto**

- Sistema Control de Sacramentos y Asignación de Personal
  - Módulo de Feligrés
  - Módulo de Sacramentos
  - Módulo de Búsqueda y Reportes de Control
  - Módulo de Personal
  - Módulo de Comunidades/Grupos
  - Módulo de Consultas y Reportes de Asignación
  - Módulo de Backup's
- Manual técnico
- Manual de usuario
- Informe de Pruebas integrales
- Plan de puesta en marcha
- Informe de Soporte

### **Gestión del Proyecto**

- Acta de Constitución del Proyecto (Project Charter)
- Plan de Gestión del Proyecto
- Plan de Gestión del Alcance
- Estructura del desglose del trabajo – EDT (WBS)
- Cronograma del proyecto
- Plan de gestión de costos del proyecto
- Plan de Gestión de Requerimientos
- Plan de Gestión de Cambios
- Plan de Gestión de la Calidad
- Plan de Gestión de riesgos
- Plan de Gestión de las comunicaciones
- Plan de Gestión de adquisiciones
- Plan de Gestión de RRHH
- Lecciones aprendidas
- Aceptación Final del Proyecto

## B. Tiempo

### Definición de las Actividades

EDT	ACTIVIDAD	DESCRIPCIÓN
1	<b>INICIO</b>	
1.1	<b>Modelamiento de Negocio</b>	
1.1.1	Reunión con el Párroco	Reunión con el padre Vicente para obtener permiso para ejecutar el proyecto
1.1.2	Reunión con la Secretaría	Reunión con la secretaría para obtener información general sobre las actividades y problemas
1.1.3	Obtención de Documentación	Obtener las plantillas de actas de sacramentos y de registro de los mismos
1.1.4	Análisis de Documentos	Analizar la documentación para encontrar y clasificar los datos
1.1.5	Elaborar Diagrama de Caso de Uso de Negocio	Elaborar los diagramas del funcionamiento de la secretaría
1.2	<b>Requisitos</b>	
1.2.1	Reunión con la Secretaría y el Párroco	Segunda reunión con ambos interesados para confirmar el análisis de la información y añadir nuevo información
1.2.2	Elaboración de Requisitos	Elaborar los requisitos que tendrá el sistema
2	<b>ELABORACION</b>	
2.1	Análisis de Documentos	Analizar la documentación obtenida para determinar qué datos se usarán
2.2	Análisis de Requisitos	Analizar los requisitos y determinar el alcance del sistema a elaborar
2.3	<b>Diseño de Diagramas</b>	
2.3.1	Elaborar Diagrama de Caso de Uso de Sistema	Elaborar los diagramas que explican de manera genérica las funciones del sistema
2.3.2	Elaborar Diagrama de Secuencia, Paquetes y Despliegue	Elaborar los diagramas que explican cómo se ejecutarán las funciones principales dentro del sistema
2.3.3	Elaborar Diagrama de Clases	Elaborar el diagrama del modelo de datos del sistema
2.4	Modelar Base de Datos	Elaborar el prototipo de la base de datos del sistema
2.5	<b>Diseñar Prototipo de Interfaces</b>	
2.5.1	Elaborar Prototipos	Elaborar diseños tentativos de las interfaces del sistema y la distribución de los datos
2.5.2	Verificar Prototipos con la Secretaría	Presentar los prototipos al líder de usuario
2.5.3	Ejecutar correcciones	Corregir los prototipos para su mejora
2.5.4	Obtención de conformidad de prototipos	Conseguir aprobación de los prototipos de las interfaces

<b>3</b>	<b>CONSTRUCCION</b>	
3.1	Implementar Base de Datos en MySql	Implantar la base de datos en el gestor de base de datos MySql
3.2	Diseñar Interfaces del Sistema	Diseñar las interfaces del sistema usando las herramientas establecidas
<b>3.3</b>	<b>Sub-Sistema de Control de Sacramentos</b>	
3.3.1	Codificar Módulo de Feligrés	Codificar las interfaces del módulo de feligrés
3.3.2	Ejecutar Pruebas del Módulo	Probar el funcionamiento de las interfaces del módulo
3.3.3	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
3.3.4	Codificar Módulo de Sacramentos	Codificar las interfaces del módulo de sacramentos
3.3.5	Ejecutar Pruebas del Módulo	Probar el funcionamiento de las interfaces del módulo
3.3.6	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
3.3.7	Codificar Módulo de Reportes de Sacramentos	Codificar la interfaz de impresión de actas
3.3.8	Ejecutar Pruebas del Módulo	Probar el funcionamiento de la interfaz
3.3.9	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
<b>3.4</b>	<b>Sub-Sistema de Asignación de Personal</b>	
3.4.1	Codificar Módulo de Personal	Codificar las interfaces del módulo de personal
3.4.2	Ejecutar Pruebas del Módulo	Probar el funcionamiento de las interfaces del módulo
3.4.3	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
3.4.4	Codificar Módulo de Comunidades	Codificar las interfaces del módulo de comunidades
3.4.5	Ejecutar Pruebas del Módulo	Probar el funcionamiento de las interfaces del módulo
3.4.6	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
3.4.7	Codificar Módulo de Reportes de Asignación	Codificar la interfaz de reportes de asignación de personal
3.4.8	Ejecutar Pruebas del Módulo	Probar el funcionamiento de la interfaz
3.4.9	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
3.5	Codificar Módulo de Backups	Codificar las interfaces del módulo de backup's
3.6	Ejecutar Pruebas del Módulo	Probar el funcionamiento de las interfaces del módulo
3.7	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
3.8	Elaborar Manual Técnico	Elaborar Diccionario de Datos
3.9	Elaborar Manual de Usuario	Elaborar Manual de uso del sistema
<b>4</b>	<b>TRANSICION</b>	
4.1	Ejecutar Pruebas Integrales	Probar todo el sistema en

		conjunto
4.2	Corrección de Errores Detectados	Corregir las posibles fallas detectadas en las pruebas
4.3	Elaborar Acta de Pruebas	Elaborar acta de pruebas realizadas y su estado
4.4	Implantación del Sistema en la Parroquia	Instalar las versiones finales de la base de datos y del sistema en la máquina de la secretaría
4.5	Capacitación a los Usuarios	Capacitar a la secretaría y al párroco en el uso de todo el sistema
4.6	Entrega de Manuales	Entregar los manuales técnico y usuario a la parroquia
4.7	Elaborar Acta de Aceptación	Elaborar el acta de aceptación del sistema
<b>5</b>	<b>GESTION DEL PROYECTO</b>	
<b>5.1</b>	<b>INICIO</b>	
<b>1.5.1.1</b>	<b>Elaborar Acta de Constitución</b>	
1.5.1.1.1	Reunion con el Párroco y la Secretaria	Reunión con el padre Vicente y la secretaría para obtener permiso de ejecutar el proyecto y obtener información
1.5.1.1.2	Elaborar Acta	Elaborar el documento donde se explican los fundamentos del inicio del proyecto
1.5.1.2	Identificar Interesados	Identificar a las personas involucradas en el proyecto
<b>5.2</b>	<b>PLANIFICACION</b>	
5.2.1	Elaborar Plan de Gestión de Py	Elaborar el documento de dirección del proyecto
5.2.2	Elaborar Plan de Gestión de Cambios	Elaborar el documento de gestión de cambios
5.2.3	Elaborar Plan de Gestión de Requerimientos	Elaborar el documento de gestión de requerimientos
5.2.4	Elaborar Plan de Gestión de Alcance	Elaborar el documento de gestión de alcance
5.2.5	Elaborar EDT	Elaborar el EDT (Estructura de Desglose de Trabajo)
5.2.6	Elaborar Cronograma	Elaborar el cronograma de actividades
5.2.7	Elaborar Presupuesto	Elaborar el presupuesto del proyecto
5.2.8	Elaborar Plan de Gestión de Calidad del Py	Elaborar el documento de gestión de calidad del proyecto
5.2.9	Elaborar Plan de Gestión de Comunicaciones	Elaborar el documento de gestión de comunicaciones
5.2.10	Elaborar Plan de Gestión de Riesgos	Elaborar de documento de gestión de riesgos
<b>5.2.11</b>	<b>Elaborar Plan de Gestión de RRHH</b>	
5.2.11.1	Matriz RAM	Elaborar la matriz RAM (Manejo de Asignación de Responsabilidades)
5.2.11.2	Plan de Gestión de RRHH	Elaborar el documento de gestión de recursos humanos
5.2.12	Elaborar Plan de Gestión de Adquisiciones	Elaborar el documento de gestión de adquisiciones

<b>5.3</b>	<b>EJECUCION</b>	
5.3.1	Dirigir y Gestionar la ejecución del Py	Dirigir la ejecución del proyecto
<b>5.3.2</b>	<b>Aseguramiento de la Calidad</b>	
5.3.2.1	Documentar cambios solicitados	Documentar los posibles cambios en el proyecto
5.3.2.2	Actualizar plan de gestión del Py	Actualizar a una nueva versión el documento de gestión del proyecto
5.3.3	Definir Acciones Correctivas recomendadas	Determinar qué acciones se tomarán ante el cambio
5.3.4	Realizar la distribución de la información	Enviar la actualización de la documentación al párroco y líder de usuario
<b>5.4</b>	<b>SUPERVISION Y CONTROL</b>	
<b>5.4.1</b>	<b>Supervisar y controlar el trabajo del Py</b>	
5.4.1.1	Documentar las acciones correctivas recomendadas	Elaborar los documentos de acciones correctivas recomendadas
5.4.1.2	Documentar las acciones preventivas recomendadas	Elaborar los documentos de acciones preventivas recomendadas
5.4.1.3	Documentar cambios solicitados	Documentar las solicitudes de cambio
<b>5.4.2</b>	<b>Control Integrado de Cambios</b>	
5.4.2.1	Documentar solicitudes de cambio solicitadas	Documentar las solicitudes de cambio
5.4.2.2	Documentar solicitudes de cambio rechazadas	Documentar las solicitudes de cambio rechazadas
5.4.2.3	Documentar acciones correctivas aprobadas	Documentar las acciones correctivas aprobadas
5.4.2.4	Documentar acciones preventivas aprobadas	Documentar las acciones preventivas aprobadas
5.4.2.5	Definir los Productos Entregables	Determinar los entregables del proyecto
<b>5.4.3</b>	<b>Verificación del Alcance</b>	
5.4.3.1	Documentar los productos entregables aceptados	Documentar los entregables aceptados
5.4.3.2	Documentar los cambios solicitados	Documentar las solicitudes de cambio
5.4.3.3	Documentar las acciones correctivas recomendadas	Elaborar los documentos de acciones correctivas recomendadas
<b>5.4.4</b>	<b>Control del Alcance</b>	
5.4.4.1	Documentar los cambios solicitados	Documentar las solicitudes de cambio
5.4.4.2	Documentar las acciones correctivas recomendadas	Elaborar los documentos de acciones correctivas recomendadas
<b>5.4.5</b>	<b>Control del Cronograma</b>	
5.4.5.1	Documentar los cambios solicitados	Documentar las solicitudes de cambio
5.4.5.2	Documentar las acciones correctivas recomendadas	Elaborar los documentos de acciones correctivas recomendadas
<b>5.4.6</b>	<b>Control de Costos</b>	

5.4.6.1	Documentar la conclusión proyectada	Documentar los costos proyectados
5.4.6.2	Documentar los cambios solicitados	Documentar las solicitudes de cambio
5.4.6.3	Documentar las acciones correctivas recomendadas	Elaborar los documentos de acciones correctivas recomendadas
<b>5.4.7</b>	<b>Control y Seguimiento de Riesgos</b>	
5.4.7.1	Documentar riesgos surgidos	Documentar los riesgos que se llevaron a cabo durante el proyecto
5.4.7.2	Documentar acciones realizadas	Documentar las acciones realizadas frente a los riesgos surgidos
<b>5.4.8</b>	<b>Control de Calidad</b>	
5.4.8.1	Documentar la reparación de defectos recomendada	Documentar las correcciones
5.4.8.2	Documentar los productos entregables validados	Documentar los entregables validados por el párroco y la secretaría
5.4.8.3	Definir Acciones Preventivas recomendadas	Determinar las acciones preventivas recomendadas
<b>5.5</b>	<b>CIERRE</b>	
<b>5.5.1</b>	<b>Cierre del Proyecto</b>	
<b>5.5.1.1</b>	<b>Cierre Administrativo</b>	
<b>5.5.1.1.1</b>	<b>Lecciones Aprendidas</b>	
5.5.1.1.1.1	Documentar Lecciones Aprendidas	Documentar las lecciones aprendidas durante el proyecto
<b>5.5.2</b>	<b>Cierre del Contrato</b>	
<b>5.5.2.1</b>	<b>Aceptación Final del Proyecto</b>	
5.5.2.1.1	Entregar Producto Final	Entregar el sistema a la parroquia

Tabla 3 Definición de las actividades

## Hitos

RESUMEN DE HITOS	FECHA ESPERADA
Entrega del primer subsistema: Sistema de Control de Sacramentos	11/09/2013
Entrega del Segundo Subsistema: Sistema de Asignación de Personal	16/10/2013
Pruebas Integrales: Sistema de Control de Sacramentos y Asignación de Personal	11/11/2013
Capacitación: Sistema de Control de Sacramentos y Asignación de Personal	19/11/2013
Cierre del Contrato	21/11/2013

Tabla 4 Hitos del Proyecto

**Cuadro de los responsables de las tareas**

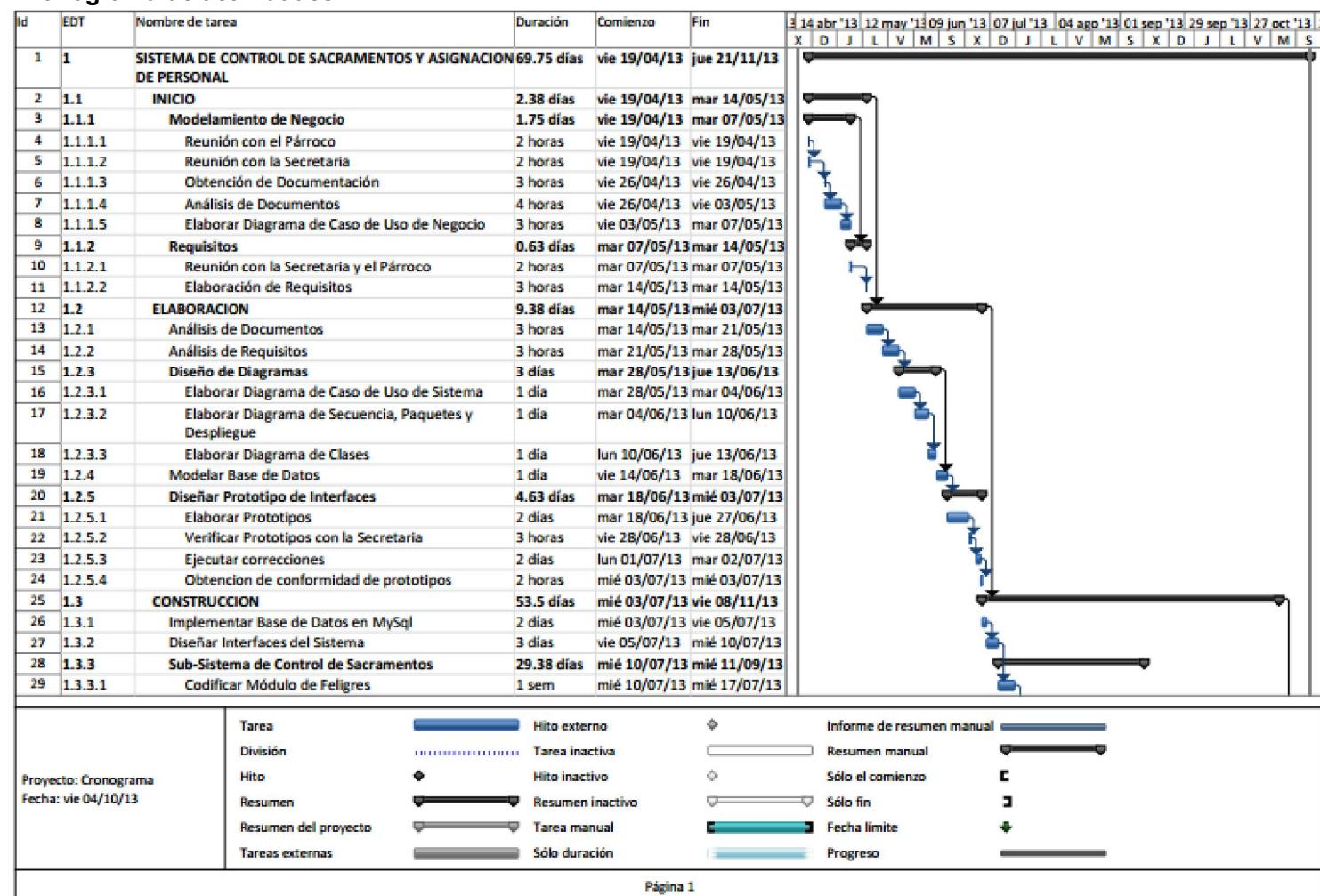
<b>EDT</b>	<b>ACTIVIDAD</b>	<b>RESPONSABLES</b>
<b>1</b>	<b>INICIO</b>	
<b>1.1</b>	<b>Modelamiento de Negocio</b>	
1.1.1	Reunión con el Párroco	JUAN CASTILLO
1.1.2	Reunión con la Secretaria	JUAN CASTILLO
1.1.3	Obtención de Documentación	JUAN CASTILLO
1.1.4	Análisis de Documentos	JUAN CASTILLO
1.1.5	Elaborar Diagrama de Caso de Uso de Negocio	JUAN CASTILLO
<b>1.2</b>	<b>Requisitos</b>	
1.2.1	Reunión con la Secretaria y el Párroco	JUAN CASTILLO
1.2.2	Elaboración de Requisitos	JUAN CASTILLO
<b>2</b>	<b>ELABORACION</b>	
2.1	Análisis de Documentos	JUAN CASTILLO
2.2	Análisis de Requisitos	JUAN CASTILLO
<b>2.3</b>	<b>Diseño de Diagramas</b>	
2.3.1	Elaborar Diagrama de Caso de Uso de Sistema	JUAN CASTILLO
2.3.2	Elaborar Diagrama de Secuencia, Paquetes y Despliegue	JUAN CASTILLO
2.3.3	Elaborar Diagrama de Clases	JUAN CASTILLO
2.4	Modelar Base de Datos	JUAN CASTILLO
<b>2.5</b>	<b>Diseñar Prototipo de Interfaces</b>	
2.5.1	Elaborar Prototipos	JUAN CASTILLO
2.5.2	Verificar Prototipos con la Secretaria	JUAN CASTILLO
2.5.3	Ejecutar correcciones	JUAN CASTILLO
2.5.4	Obtención de conformidad de prototipos	JUAN CASTILLO
<b>3</b>	<b>CONSTRUCCION</b>	
3.1	Implementar Base de Datos en MySql	JUAN CASTILLO
3.2	Diseñar Interfaces del Sistema	JUAN CASTILLO
<b>3.3</b>	<b>Sub-Sistema de Control de Sacramentos</b>	
3.3.1	Codificar Módulo de Feligrés	JUAN CASTILLO
3.3.2	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.3.3	Elaborar Acta de Pruebas	JUAN CASTILLO
3.3.4	Codificar Módulo de Sacramentos	JUAN CASTILLO
3.3.5	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.3.6	Elaborar Acta de Pruebas	JUAN CASTILLO
3.3.7	Codificar Módulo de Reportes de Sacramentos	JUAN CASTILLO
3.3.8	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.3.9	Elaborar Acta de Pruebas	JUAN CASTILLO
<b>3.4</b>	<b>Sub-Sistema de Asignación de Personal</b>	
3.4.1	Codificar Módulo de Personal	JUAN CASTILLO
3.4.2	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.4.3	Elaborar Acta de Pruebas	JUAN CASTILLO
3.4.4	Codificar Módulo de Comunidades	JUAN CASTILLO
3.4.5	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.4.6	Elaborar Acta de Pruebas	JUAN CASTILLO

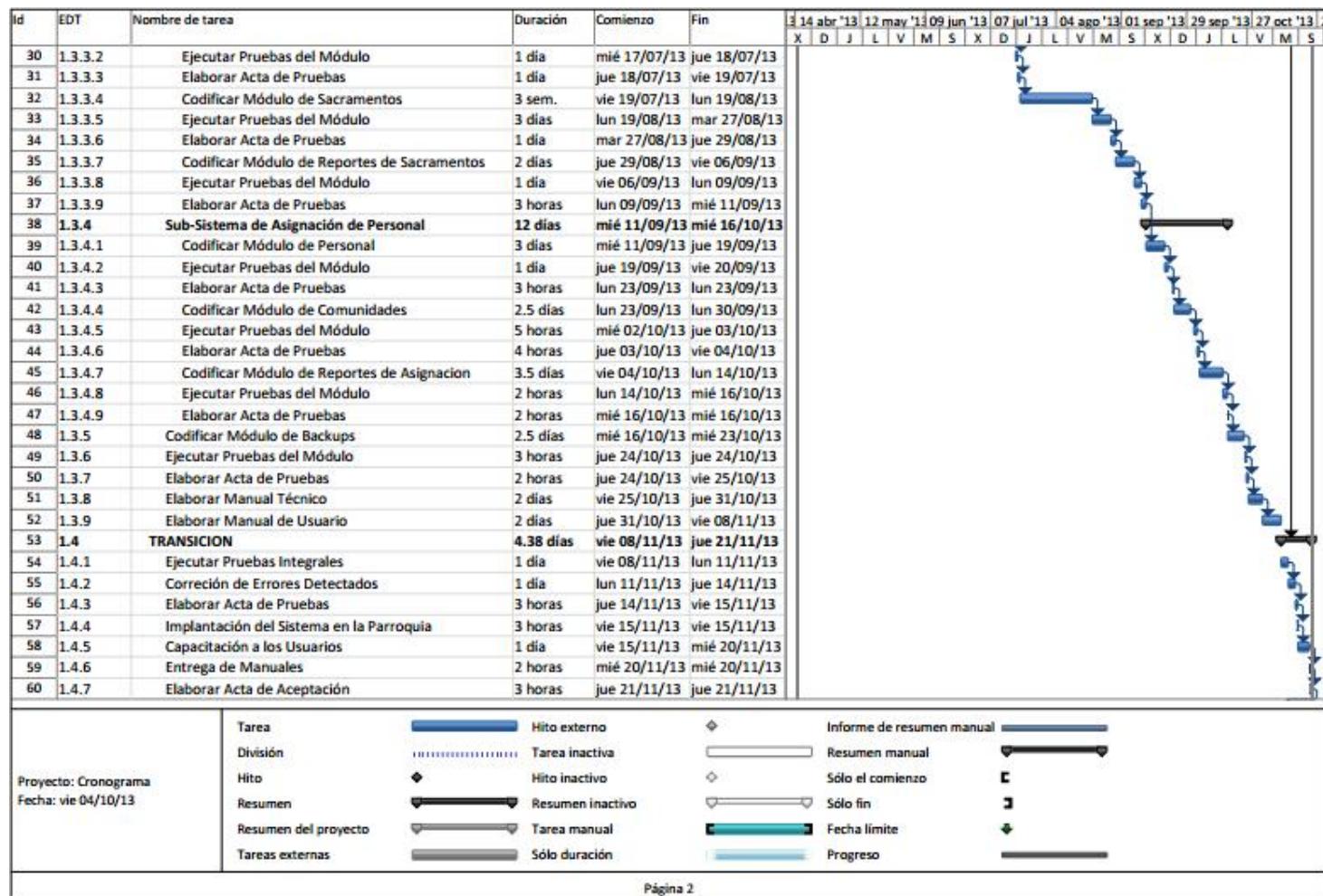
3.4.7	Codificar Módulo de Reportes de Asignación	JUAN CASTILLO
3.4.8	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.4.9	Elaborar Acta de Pruebas	JUAN CASTILLO
3.5	Codificar Módulo de Backups	JUAN CASTILLO
3.6	Ejecutar Pruebas del Módulo	JUAN CASTILLO, LUISA CAJO
3.7	Elaborar Acta de Pruebas	JUAN CASTILLO
3.8	Elaborar Manual Técnico	JUAN CASTILLO
3.9	Elaborar Manual de Usuario	JUAN CASTILLO
<b>4</b>	<b>TRANSICION</b>	
4.1	Ejecutar Pruebas Integrales	JUAN CASTILLO, LUISA CAJO, LUIS LOO
4.2	Corrección de Errores Detectados	JUAN CASTILLO
4.3	Elaborar Acta de Pruebas	JUAN CASTILLO
4.4	Implantación del Sistema en la Parroquia	JUAN CASTILLO
4.5	Capacitación a los Usuarios	JUAN CASTILLO
4.6	Entrega de Manuales	JUAN CASTILLO
4.7	Elaborar Acta de Aceptación	JUAN CASTILLO
<b>5</b>	<b>GESTION DEL PROYECTO</b>	
<b>5.1</b>	<b>INICIO</b>	
5.1.1	Elaborar Acta de Constitución	JUAN CASTILLO
5.1.1.1	Reunión con el Párroco y la Secretaría	JUAN CASTILLO
5.1.1.2	Elaborar Acta	JUAN CASTILLO
5.1.2	Identificar Interesados	JUAN CASTILLO
<b>5.2</b>	<b>PLANIFICACION</b>	
5.2.1	Elaborar Plan de Gestión de Py	JUAN CASTILLO
5.2.2	Elaborar Plan de Gestión de Cambios	JUAN CASTILLO
5.2.3	Elaborar Plan de Gestión de Requerimientos	JUAN CASTILLO
5.2.4	Elaborar Plan de Gestión de Alcance	JUAN CASTILLO
5.2.5	Elaborar EDT	JUAN CASTILLO
5.2.6	Elaborar Cronograma	JUAN CASTILLO
5.2.7	Elaborar Presupuesto	JUAN CASTILLO
5.2.8	Elaborar Plan de Gestión de Calidad del Py	JUAN CASTILLO
5.2.9	Elaborar Plan de Gestión de Comunicaciones	JUAN CASTILLO
5.2.10	Elaborar Plan de Gestión de Riesgos	JUAN CASTILLO
<b>5.2.11</b>	<b>Elaborar Plan de Gestión de RRHH</b>	
5.2.11.1	Matriz RAM	JUAN CASTILLO
5.2.11.2	Plan de Gestión de RRHH	JUAN CASTILLO
5.2.12	Elaborar Plan de Gestión de Adquisiciones	JUAN CASTILLO
<b>5.3</b>	<b>EJECUCION</b>	JUAN CASTILLO
5.3.1	Dirigir y Gestionar la ejecución del Py	JUAN CASTILLO
<b>5.3.2</b>	<b>Aseguramiento de la Calidad</b>	
5.3.2.1	Documentar cambios solicitados	JUAN CASTILLO
5.3.2.2	Actualizar plan de gestión del Py	JUAN CASTILLO
5.3.3	Definir Acciones Correctivas recomendadas	JUAN CASTILLO
5.3.4	Realizar la distribución de la información	JUAN CASTILLO
<b>5.4</b>	<b>SUPERVISION Y CONTROL</b>	

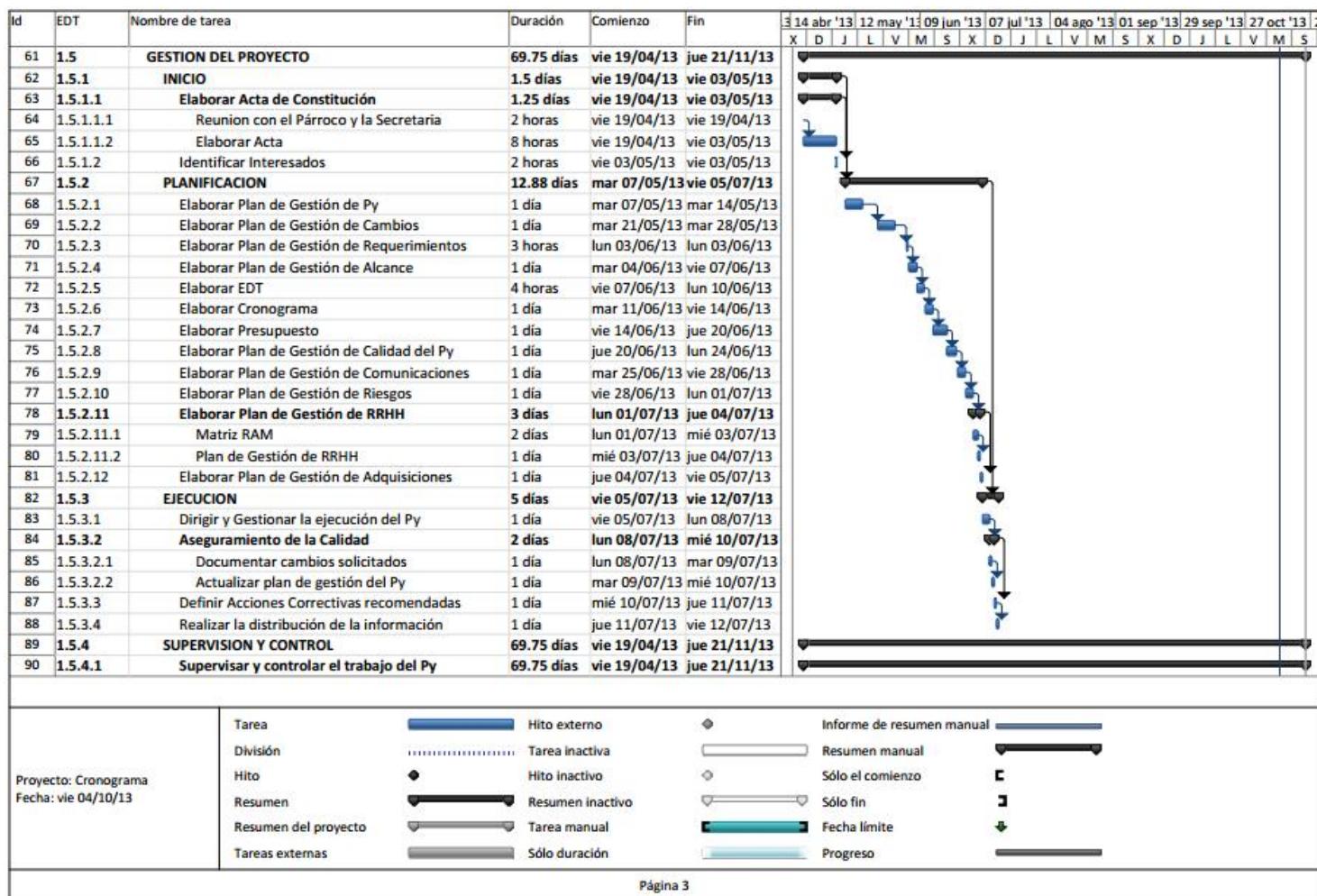
<b>5.4.1</b>	<b>Supervisar y controlar el trabajo del Py</b>	
5.4.1.1	Documentar las acciones correctivas recomendadas	JUAN CASTILLO
5.4.1.2	Documentar las acciones preventivas recomendadas	JUAN CASTILLO
5.4.1.3	Documentar cambios solicitados	JUAN CASTILLO
<b>5.4.2</b>	<b>Control Integrado de Cambios</b>	
5.4.2.1	Documentar solicitudes de cambio solicitadas	JUAN CASTILLO
5.4.2.2	Documentar solicitudes de cambio rechazadas	JUAN CASTILLO
5.4.2.3	Documentar acciones correctivas aprobadas	JUAN CASTILLO
5.4.2.4	Documentar acciones preventivas aprobadas	JUAN CASTILLO
5.4.2.5	Definir los Productos Entregables	JUAN CASTILLO
<b>5.4.3</b>	<b>Verificación del Alcance</b>	
5.4.3.1	Documentar los productos entregables aceptados	JUAN CASTILLO
5.4.3.2	Documentar los cambios solicitados	JUAN CASTILLO
5.4.3.3	Documentar las acciones correctivas recomendadas	JUAN CASTILLO
<b>5.4.4</b>	<b>Control del Alcance</b>	
5.4.4.1	Documentar los cambios solicitados	JUAN CASTILLO
5.4.4.2	Documentar las acciones correctivas recomendadas	JUAN CASTILLO
<b>5.4.5</b>	<b>Control del Cronograma</b>	
5.4.5.1	Documentar los cambios solicitados	JUAN CASTILLO
5.4.5.2	Documentar las acciones correctivas recomendadas	JUAN CASTILLO
<b>5.4.6</b>	<b>Control de Costos</b>	
5.4.6.1	Documentar la conclusión proyectada	JUAN CASTILLO
5.4.6.2	Documentar los cambios solicitados	JUAN CASTILLO
5.4.6.3	Documentar las acciones correctivas recomendadas	JUAN CASTILLO
<b>5.4.7</b>	<b>Control y Seguimiento de Riesgos</b>	
5.4.7.1	Documentar riesgos surgidos	JUAN CASTILLO
5.4.7.2	Documentar acciones realizadas	JUAN CASTILLO
<b>5.4.8</b>	<b>Control de Calidad</b>	
5.4.8.1	Documentar la reparación de defectos recomendada	JUAN CASTILLO, LUIS LOO
5.4.8.2	Documentar los productos entregables validados	JUAN CASTILLO, LUIS LOO
5.4.8.3	Definir Acciones Preventivas recomendadas	JUAN CASTILLO, LUIS LOO
<b>5.5</b>	<b>CIERRE</b>	
<b>5.5.1</b>	<b>Cierre del Proyecto</b>	
<b>5.5.1.1</b>	<b>Cierre Administrativo</b>	
<b>5.5.1.1.1</b>	<b>Lecciones Aprendidas</b>	
5.5.1.1.1.1	Documentar Lecciones Aprendidas	JUAN CASTILLO
<b>5.5.2</b>	<b>Cierre del Contrato</b>	
<b>5.5.2.1</b>	<b>Aceptación Final del Proyecto</b>	
5.5.2.1.1	Entregar Producto Final	JUAN CASTILLO

Tabla 5 Cuadro de los responsables de las tareas

## Cronograma de actividades







Id	EDT	Nombre de tarea	Duración	Comienzo	Fin	Calendario														
						31 abr '13	12 may '13	09 jun '13	07 jul '13	04 ago '13	01 sep '13	29 sep '13	27 oct '13	X	D	J	L	V	M	S
91	1.5.4.1.1	Documentar las acciones correctivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
92	1.5.4.1.2	Documentar las acciones preventivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
93	1.5.4.1.3	Documentar cambios solicitados	69.75 días	vie 19/04/13	jue 21/11/13															
94	1.5.4.2	<b>Control Integrado de Cambios</b>	69.75 días	vie 19/04/13	jue 21/11/13															
95	1.5.4.2.1	Documentar solicitudes de cambio solicitadas	69.75 días	vie 19/04/13	jue 21/11/13															
96	1.5.4.2.2	Documentar solicitudes de cambio rechazadas	69.75 días	vie 19/04/13	jue 21/11/13															
97	1.5.4.2.3	Documentar acciones correctivas aprobadas	69.75 días	vie 19/04/13	jue 21/11/13															
98	1.5.4.2.4	Documentar acciones preventivas aprobadas	69.75 días	vie 19/04/13	jue 21/11/13															
99	1.5.4.2.5	Definir los Productos Entregables	69.75 días	vie 19/04/13	jue 21/11/13															
100	1.5.4.3	<b>Verificación del Alcance</b>	69.75 días	vie 19/04/13	jue 21/11/13															
101	1.5.4.3.1	Documentar los productos entregables aceptados	69.75 días	vie 19/04/13	jue 21/11/13															
102	1.5.4.3.2	Documentar los cambios solicitados	69.75 días	vie 19/04/13	jue 21/11/13															
103	1.5.4.3.3	Documentar las acciones correctivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
104	1.5.4.4	<b>Control del Alcance</b>	69.75 días	vie 19/04/13	jue 21/11/13															
105	1.5.4.4.1	Documentar los cambios solicitados	69.75 días	vie 19/04/13	jue 21/11/13															
106	1.5.4.4.2	Documentar las acciones correctivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
107	1.5.4.5	<b>Control del Cronograma</b>	69.75 días	vie 19/04/13	jue 21/11/13															
108	1.5.4.5.1	Documentar los cambios solicitados	69.75 días	vie 19/04/13	jue 21/11/13															
109	1.5.4.5.2	Documentar las acciones correctivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
110	1.5.4.6	<b>Control de Costos</b>	69.75 días	vie 19/04/13	jue 21/11/13															
111	1.5.4.6.1	Documentar la conclusión proyectada	69.75 días	vie 19/04/13	jue 21/11/13															
112	1.5.4.6.2	Documentar los cambios solicitados	69.75 días	vie 19/04/13	jue 21/11/13															
113	1.5.4.6.3	Documentar las acciones correctivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
114	1.5.4.7	<b>Control y Seguimiento de Riesgos</b>	69.75 días	vie 19/04/13	jue 21/11/13															
Proyecto: Cronograma Fecha: vie 04/10/13		Tarea	<div style="width: 100%; background-color: #0070C0; height: 10px;"></div>	Hito externo	<input type="radio"/>	Informe de resumen manual	<div style="width: 100%; background-color: #0070C0; height: 10px;"></div>													
		División	<div style="width: 100%; background-color: #CCCCCC; height: 10px;"></div>	Tarea Inactiva	<input type="radio"/>	Resumen manual	<div style="width: 100%; background-color: #0070C0; height: 10px;"></div>													
		Hito	<input checked="" type="radio"/>	Hito inactivo	<input type="radio"/>	Sólo el comienzo	<input type="checkbox"/>													
		Resumen	<div style="width: 100%; background-color: #0070C0; height: 10px;"></div>	Resumen Inactivo	<input type="radio"/>	Sólo fin	<input type="checkbox"/>													
		Resumen del proyecto	<div style="width: 100%; background-color: #CCCCCC; height: 10px;"></div>	Tarea manual	<input type="radio"/>	Fecha límite	<input type="checkbox"/>													
		Tareas externas	<div style="width: 100%; background-color: #CCCCCC; height: 10px;"></div>	Sólo duración	<input type="radio"/>	Progreso	<div style="width: 100%; background-color: #0070C0; height: 10px;"></div>													

Id	EDT	Nombre de tarea	Duración	Comienzo	Fin	Cronograma												
						31 abr '13	14 abr '13	12 may '13	09 jun '13	06 jul '13	03 ago '13	01 sep '13	29 sep '13	27 oct '13	03 nov '13	10 nov '13	17 nov '13	
X	D	J	L	V	M	S	X	D	J	L	V	M	S	X	D	J	V	M
115	1.5.4.7.1	Documentar riesgos surgidos	69.75 días	vie 19/04/13	jue 21/11/13													
116	1.5.4.7.2	Documentar acciones realizadas	69.75 días	vie 19/04/13	jue 21/11/13													
117	1.5.4.8	<b>Control de Calidad</b>	69.75 días	vie 19/04/13	jue 21/11/13													
118	1.5.4.8.1	Documentar la reparación de defectos recomendada	69.75 días	vie 19/04/13	jue 21/11/13													
119	1.5.4.8.2	Documentar los productos entregables validados	69.75 días	vie 19/04/13	jue 21/11/13													
120	1.5.4.8.3	Definir Acciones Preventivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13													
121	1.5.5	<b>CIERRE</b>	2 días	vie 15/11/13	jue 21/11/13													
122	1.5.5.1	<b>Cierre del Py</b>	1 día	vie 15/11/13	lun 18/11/13													
123	1.5.5.1.1	<b>Cierre Administrativo</b>	1 día	vie 15/11/13	lun 18/11/13													
124	1.5.5.1.1.1	<b>Lecciones Aprendidas</b>	1 día	vie 15/11/13	lun 18/11/13													
125	1.5.5.1.1.1.	Documentar Lecciones Aprendidas	1 día	vie 15/11/13	lun 18/11/13													
126	1.5.5.2	<b>Cierre del Contrato</b>	1 día	lun 18/11/13	jue 21/11/13													
127	1.5.5.2.1	<b>Aceptación Final del Py</b>	1 día	lun 18/11/13	jue 21/11/13													
128	1.5.5.2.1.1	Entregar Producto Final	1 día	lun 18/11/13	jue 21/11/13													
<hr/>																		
Proyecto: Cronograma Fecha: vie 04/10/13		Tarea		Hito externo		◆		Informe de resumen manual										
		División		Tarea inactiva		◆		Resumen manual										
		Hito	◆	Hito inactivo	◆	◆		Sólo el comienzo	■									
		Resumen	◆	Resumen inactivo	◆	◆	▼	Sólo fin	■									
		Resumen del proyecto	◆	Tarea manual	◆	◆	▼	Fecha límite	▼									
		Tareas externas	◆	Sólo duración	◆	◆	▼	Progreso	▼									

**C. Costo****Línea Base**

S/. 11,878.00

**Reserva**

S/. 1,187.80

**Total**

S/. 13,065.80

**Identificación de recursos****Materiales**

- 2 Millares de Hojas Bond A-4 75 gramos
- 3 Pomo de Tinta Negra
- 1 Pomo de Tinta Azul
- 1 Pomo de Tinta Amarillo
- 1 Pomo de Tinta Rojo
- 1 Lapicero PILOT Tinta Negra
- 1 Lapicero PILOT Tinta Azul
- 5 Folder Manila A-4
- 2 Archivadores
- 1 Perforadora de Papel
- 20 Micas Portapapeles

**Equipos**

- 1 PC Escritorio As-Rock NS-64 300 GB, 2 GB RAM
- 1 Impresora Multifuncional Canon MP 495
- 1 Memoria USB 8 GB

**Servicios**

- Internet
- Movilidad y viáticos
- Alquiler de proyectos multimedia

**Recurso Humano**

- 1 Jefe de Proyecto
- 1 Analista Funcional
- 1 Programador Senior
- 1 Programador Junior
- 1 Analista de Calidad
- 1 Administrador de Base de Datos

## Plan de Gestión para Costos

PLAN DE GESTION DE COSTOS	PROY-001															
<b>Componente</b>	Ver 1.0 20/06/2013															
<b>Objetivo:</b>  Determinar la estructura de costos necesaria del Proyecto <b>SISTEMA DE CONTROL DE SACRAMENTOS Y ASIGNACIÓN DE PERSONAL</b> para su seguimiento y control.																
<b>Roles y Responsabilidades</b>  <ul style="list-style-type: none"> <li>- Jefe de Proyecto Responsable de coordinar con los interesados, elaborar el cronograma, efectuar el seguimiento al cronograma e informar a nivel superior de las posibles solicitudes de cambios, así como del normal desarrollo del Proyecto.</li> <li>- Analista Funcional Responsable de establecer las definiciones funcionales del Sistema, elaborar las solicitudes de control de cambios y efectuar el seguimiento a los programadores.</li> <li>- Programador Senior Responsable de elaborar la programación del sistema, de emplear los estándares de programación y de asignarle tareas al programador junior.</li> <li>- Programador Junior Responsable de elaborar los diagramas y programación.</li> <li>- Analista de Calidad Responsable de verificar la optimización de los programas así como el cumplimiento a los estándares de programación.</li> <li>- Administrador de Base de Datos Responsable de verificar la correcta normalización de tablas y la creación debida de tablas y campos.</li> </ul>																
<b>Presupuesto del proyecto</b>  El presupuesto aprobado es de S/. 13,065.80 con un margen del 5% En términos de trabajo:	<table> <thead> <tr> <th></th> <th>Duración</th> <th>Costo</th> </tr> </thead> <tbody> <tr> <td>SISTEMA IMPLEMENTADO</td> <td>69.75 días</td> <td>S/. 11,878.00</td> </tr> <tr> <td>- Gestión</td> <td>2.38 días</td> <td>S/. 1,017.00</td> </tr> <tr> <td>- Logística</td> <td>3 días</td> <td>S/. 2,547.00</td> </tr> <tr> <td>- Elaboración</td> <td>53.5 días</td> <td>S/. 6,705.00</td> </tr> </tbody> </table>		Duración	Costo	SISTEMA IMPLEMENTADO	69.75 días	S/. 11,878.00	- Gestión	2.38 días	S/. 1,017.00	- Logística	3 días	S/. 2,547.00	- Elaboración	53.5 días	S/. 6,705.00
	Duración	Costo														
SISTEMA IMPLEMENTADO	69.75 días	S/. 11,878.00														
- Gestión	2.38 días	S/. 1,017.00														
- Logística	3 días	S/. 2,547.00														
- Elaboración	53.5 días	S/. 6,705.00														

- Calidad	6.49 días	S/. 977.00
- Resultados	4.38 días	S/. 632.0
En términos de recursos:		
- Jefe de Proyecto	RRHH	S/. 2,420.00
- Analista Funcional	RRHH	S/. 924.00
- Programador Senior	RRHH	S/. 2,583.00
- Programador Junior	RRHH	S/. 1,456.00
- Analista de Calidad	RRHH	S/. 1,080.00
- Administrador de Base de Datos	RRHH	S/. 228.00
- PC de Escritorio	SERV	S/. 2,000.00
- Impresora Multifuncional	SERV	S/. 320.00
- USB 8GB	SERV	S/. 45.00
- Internet	SERV	S/. 400.00
- Movilidad	SERV	S/. 160.00
- Alquiler de Proyector	SERV	S/. 80.00
- Materiales de Oficina	MATERIAL	S/. 182.00
 PRESUPUESTO BASE S/. 11,878.00		
RESERVA S/. 1,187.80		
TOTAL PRESUPUESTO S/. 13,065.80		
<b>PRESUPUESTO REAL (SIN RECURSO HUMANO) S/. 3,187.00</b>		
 <b>Criterios de costos:</b>		
<ul style="list-style-type: none"> <li>- Las personas o los integrantes estarán solamente asignadas a este Proyecto por lo tanto sus costos por remuneración son al 100%.</li> </ul>		
 <b>Mediciones de costo</b>		
<ul style="list-style-type: none"> <li>- Costo Efectivo / Costo Programado</li> </ul> <p>Nos indicará si estamos dentro de los límites establecidos a nivel de costos. Este indicador deberá ser &lt; 1 o igual a 1.</p> <p>Este indicador se llevará en forma mensual</p> <ul style="list-style-type: none"> <li>- Costo no programado / Costo presupuestado restante</li> </ul> <p>Nos indicará que porcentaje del total presupuestado restante a la fecha estamos utilizándolo, de tal forma que se puede decidir si es necesario ampliar el presupuesto inicial o tomar medidas de ajuste a nivel de costos con todo lo restante.</p>		

Tabla 6 Plan de Gestión de Costos

## Cuadro de Costos

### MATERIALES

Nº	DESCRIPCIÓN	UNIDAD DE MEDIDA	CANTIDAD	COSTO UNITARIO	SUB TOTAL		
					EFEKTIVO	VALORIZADO	IMPORTE
1	Papel bond A-4 75 gramos	Millar	2	S/. 24.00	S/. 48.00		S/. 48.00
2	Pomo Tinta Negra para impresora.	Unidad.	3	S/. 12.00	S/. 36.00	S/. -	S/. 36.00
3	Pomo Tinta Amarillo para impresora.	Unidad.	1	S/. 12.00	S/. 12.00	S/. -	S/. 12.00
4	Pomo Tinta Azul para impresora.	Unidad.	1	S/. 12.00	S/. 12.00	S/. -	S/. 12.00
5	Pomo Tinta Rojo para impresora.	Unidad.	2	S/. 12.00	S/. 12.00	S/. -	S/. 12.00
6	Lapicero PILOT Tinta Negra	Unidad.	1	S/. 3.00	S/. 3.00	S/. -	S/. 3.00
7	Lapicero PILOT Tinta Azul	Unidad.	1	S/. 3.00	S/. 3.00	S/. -	S/. 3.00
8	Folder Manila A-4	Unidad.	5	S/. 1.00	S/. 5.00	S/. -	S/. 5.00
9	Archivador	Unidad.	2	S/. 16.00	S/. 32.00	S/. -	S/. 32.00
10	Perforador	Unidad.	1	S/. 9.00	S/. -	S/. 9.00	S/. 9.00
11	Mica Portapapeles	Unidad.	20	S/. 0.50	S/. 10.00	S/. -	S/. 10.00
					<b>S/. 173.00</b>	<b>S/. 9.00</b>	<b>S/. 182.00</b>

Tabla 7 Costos de Materiales

## EQUIPOS

Nº	DESCRIPCIÓN	UNIDAD DE MEDIDA	CANTIDAD	COSTO UNITARIO	SUB TOTAL		
					EFECTIVO	VALORIZADO	IMPORTE
1	PC Escritorio As-Rock NS-64 300 GB, 2 GB RAM	Unidad.	1	S/. 2,000.00	S/. -	S/. 2,000.00	S/. 2,000.00
2	Impresora Multifuncional Canon MP 495	Unidad.	1	S/. 320.00	S/. -	S/. 320.00	S/. 320.00
3	Memoria USB 8 GB	Unidad.	1	S/. 45.00	S/. -	S/. 45.00	S/. 45.00
<b>TOTAL:</b>					S/. -	<b>S/. 2,365.00</b>	<b>S/. 2,365.00</b>

Tabla 8 Costos de Equipos

## SERVICIOS

Nº	DESCRIPCIÓN	UNIDAD DE MEDIDA	CANTIDAD	COSTO UNITARIO	SUB TOTAL		
					EFECTIVO	VALORIZADO	IMPORTE
1	Internet	Meses	8	S/. 50.00	S/. -	S/. 400.00	S/. 400.00
2	Movilidad y viáticos.	Meses	8	S/. 20.00	S/. 160.00	S/. -	S/. 160.00
3	Alquiler de proyector multimedia	Horas	8	S/. 10.00	S/. -	S/. 80.00	S/. 80.00
<b>TOTAL:</b>					<b>S/. 160.00</b>	<b>S/. 480.00</b>	<b>S/. 640.00</b>

Tabla 9 Costos de Servicios

## RECURSO HUMANO

Nº	DESCRIPCIÓN	UNIDAD DE MEDIDA	CANTIDAD	HORAS DE TRABAJO	COSTO UNITARIO	SUB TOTAL		
						EFFECTIVO	VALORIZADO	IMPORTE
1	Jefe de Proyecto	Unidad.	1	242	S/. 10.00	S/. -	S/. 2,420.00	S/. 2,420.00
2	Analista Funcional	Unidad.	1	154	S/. 6.00	S/. -	S/. 924.00	S/. 924.00
3	Programador Senior	Unidad.	1	369	S/. 7.00	S/. -	S/. 2,583.00	S/. 2,583.00
4	Programador Junior	Unidad.	1	364	S/. 4.00	S/. -	S/. 1,456.00	S/. 1,456.00
5	Analista de Calidad	Unidad.	1	180	S/. 6.00	S/. -	S/. 1,080.00	S/. 1,080.00
6	Administrador de Base de Datos	Unidad.	1	38	S/. 6.00	S/. -	S/. 228.00	S/. 228.00
<b>TOTAL:</b>						<b>S/. -</b>	<b>S/. 8,691.00</b>	<b>S/. 8,691.00</b>

Tabla 10 Costos del Recurso Humano

## RESUMEN TOTAL

Nº	DESCRIPCIÓN	SUB TOTAL		
		EFFECTIVO	VALORIZADO	IMPORTE
1	Materiales.	S/. 173.00	S/. 9.00	S/. 182.00
2	Equipos.	S/. -	S/. 2,365.00	S/. 2,365.00
3	Servicios.	S/. 160.00	S/. 480.00	S/. 640.00
<b>TOTAL REAL:</b>		<b>S/. 333.00</b>	<b>S/. 2,854.00</b>	<b>S/. 3,187.00</b>
4	Recurso Humano	S/. -	S/. 8,691.00	S/. 8,691.00
5	Reserva	S/. 1,187.80	S/. -	S/. 1,187.80
<b>TOTAL:</b>		<b>S/. 1,520.80</b>	<b>S/. 11,555.00</b>	<b>S/. 13,065.80</b>
<b>%:</b>		<b>12%</b>	<b>88%</b>	<b>100%</b>

Tabla 11 Resumen de Costos Totales

## Presupuesto

PRESUPUESTO RESUMEN DEL PROYECTO											
Nombre del Proyecto		Control de Sacramentos y Asignación de Personal							Formato		
Fecha de Creación		14/06/2013							Informe No	1	
Fecha de Última modificación		30/09/2013							AI:		
Autor		Juan Castillo							Tipo de C.		
Estado		Aprobado							Pág.	1 de 1	
Versión		1.1									
<b>Moneda en nuevos soles</b>											
EDT		Actividad	TOTAL	ABRIL	MAYO	JUNIO	JULIO	AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE
				MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8
1.1.1	Modelamiento de Negocio		S/. 182.00	S/. 86.00	S/. 96.00						
			S/. 38.00		S/. 38.00						
1.1.2	Requisitos		S/. 36.00		S/. 36.00						
			S/. 36.00		S/. 36.00						
1.2.1	Análisis de Documentos		S/. 192.00		S/. 18.00	S/. 174.00					
			S/. 96.00		S/. 96.00						
1.2.2	Análisis de Requisitos		S/. 437.00		S/. 227.00	S/. 210.00					
			S/. 437.00		S/. 227.00	S/. 210.00					
1.2.3	Diseño de Diagramas		S/. 96.00		S/. 96.00						
			S/. 96.00		S/. 96.00						
1.2.4	Modelar Base de Datos		S/. 437.00		S/. 227.00	S/. 210.00					
			S/. 437.00		S/. 227.00	S/. 210.00					
1.2.5	Diseñar Prototipo de Interfaces		S/. 437.00		S/. 227.00	S/. 210.00					
			S/. 437.00		S/. 227.00	S/. 210.00					
1.3.1	Implementar Base de Datos en MySql		S/. 437.00		S/. 227.00	S/. 210.00					
			S/. 437.00		S/. 227.00	S/. 210.00					

			362.00		362.00						
1.3.2	Diseñar Interfaces del Sistema	S/. 769.00			S/. 769.00						
1.3.3	Sub-Sistema de Control de Sacramentos	S/. 3,036.00			S/. 1,494.00	S/. 1,087.00	S/. 455.00				
1.3.4	Sub-Sistema de Asignación de Personal	S/. 1,641.00						S/. 902.50	S/. 738.50		
1.3.5	Codificar Módulo de Backups	S/. 420.00								S/. 420.00	
1.3.6	Ejecutar Pruebas del Módulo	S/. 23.00								S/. 23.00	
1.3.7	Elaborar Acta de Pruebas	S/. 42.00								S/. 42.00	
1.3.8	Elaborar Manual de Usuario	S/. 206.00								S/. 206.00	
1.3.9	Elaborar Manual Técnico	S/. 206.00						S/. 37.00	S/. 169.00		
1.4.1	Ejecutar Pruebas Integrales	S/. 53.00								S/. 53.00	
1.4.2	Corrección de Errores Detectados	S/. 288.00								S/. 288.00	
1.4.3	Elaborar Acta de Pruebas	S/. 48.00								S/. 48.00	
1.4.4	Implantación del Sistema en la Parroquia	S/. 35.00								S/. 35.00	
1.4.5	Capacitación a los Usuarios	S/.								S/. 128.00	

			128.00			
1.4.6	Entrega de Manuales	S/. 32.00				S/. 32.00
1.4.7	Elaborar Acta de Aceptación	S/. 48.00				S/. 48.00
1.5.1.1	Elaborar Acta de Constitución	S/. 112.00	S/. 72.00	S/. 40.00		
1.5.1.2	Identificar Interesados	S/. 12.00	S/. 12.00			
1.5.2.1	Elaborar Plan de Gestión de Py	S/. 80.00		S/. 80.00		
1.5.2.2	Elaborar Plan de Gestión de Cambios	S/. 80.00		S/. 80.00		
1.5.2.3	Elaborar Plan de Gestión de Requerimientos	S/. 48.00		S/. 48.00		
1.5.2.4	Elaborar Plan de Gestión de Alcance	S/. 128.00		S/. 128.00		
1.5.2.5	Elaborar EDT	S/. 24.00		S/. 24.00		
1.5.2.6	Elaborar Cronograma	S/. 48.00		S/. 48.00		
1.5.2.7	Elaborar Presupuesto	S/. 80.00		S/. 80.00		
1.5.2.8	Elaborar Plan de Gestión de Calidad del Py	S/. 128.00		S/. 128.00		

1.5.2.9	Elaborar Plan de Gestión de Comunicaciones	S/. 128.00			S/. 128.00					
1.5.2.10	Elaborar Plan de Gestión de Riesgos	S/. 80.00			S/. 10.00	S/. 70.00				
1.5.2.11	Elaborar Plan de Gestión de RRHH	S/. 240.00			S/. 240.00					
1.5.2.12	Elaborar Plan de Gestión de Adquisiciones	S/. 176.00			S/. 176.00					
1.5.3.1	Dirigir y Gestionar la ejecución del Py	S/. 176.00			S/. 176.00					
1.5.3.2	Aseguramiento de la Calidad	S/. 208.00			S/. 208.00					
1.5.3.3	Definir Acciones Correctivas recomendadas	S/. 128.00			S/. 128.00					
1.5.3.4	Realizar la distribución de la información	S/. 128.00			S/. 128.00					
1.5.4.1	Supervisar y controlar el trabajo del Py	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.4.2	Control Integrado de Cambios	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.4.3	Verificación del Alcance	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.4.4	Control del Alcance	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.4.5	Control del Cronograma	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00

1.5.4.6	Control de Costos	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.4.7	Control y Seguimiento de Riesgos	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.4.8	Control de Calidad	S/. 152.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00
1.5.5.1.1.1	Lecciones Aprendidas	S/. 176.00								S/. 176.00
1.5.5.2.1	Aceptación Final del Py	S/. 128.00								S/. 128.00
<b>Total del Costo Mensual del Proyecto</b>			S/. 322.00	S/. 576.00	S/. 1,243.00	S/. 4,113.00	S/. 1,239.00	S/. 1,509.50	S/. 1,618.50	S/. 1,257.00
<b>Total de Costo del Proyecto</b>			<b>S/. 322.00</b>	<b>S/. 898.00</b>	<b>S/. 2,141.00</b>	<b>S/. 6,254.00</b>	<b>S/. 7,493.00</b>	<b>S/. 9,002.50</b>	<b>S/. 10,621.00</b>	<b>S/. 11,878.00</b>

Tabla 12 Presupuesto Resumen del Proyecto

## D. Recursos Humanos Organización

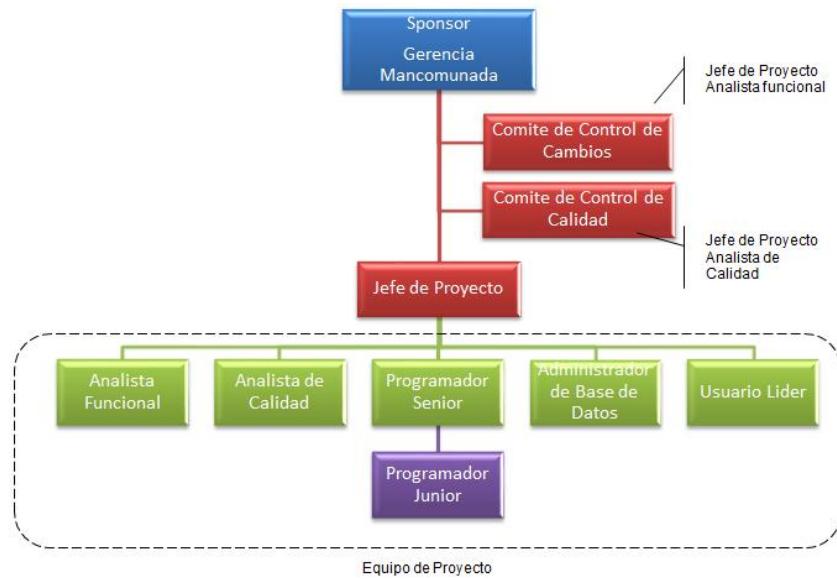


Ilustración 4 Organigrama del Equipo de Proyecto

### Matriz de asignación de responsabilidades (RAM)

ENTREGABLES	ROLES								
	SP	JP	AF	PS	PJ	AC	DBA	LU	
<b>1.0 INICIO</b>									
<b>1.1 Modelamiento de Negocio</b>									
1.1.A01 Reunión con el Párroco	P	P							
1.1.A02 Reunión con la Secretaria		P						P	
1.1.A03 Obtención de Documentación	P							P	
1.1.A04 Análisis de Documentos		R	P						
1.1.A05 Elaborar Diagrama de Caso de Uso de Negocio	A	R							
<b>1.2 Requisitos</b>									
1.2.A01 Reunión con la Secretaria y el Párroco	P	P						P	
1.2.A02 Elaboración de Requisitos	V	R							
<b>2.0 ELABORACION</b>									
2.1 Análisis de Documentos			R					P	
2.2 Análisis de Requisitos			R					P	
<b>2.3 Diseño de Diagramas</b>									
2.3.A01 Elaborar Diagrama de Caso de Uso de Sistema		A	R						
2.3.A02 Elaborar Diagrama de Secuencia, Paquetes y Despliegue		A	R						
2.3.A03 Elaborar Diagrama de Clases		A	P					R	
2.4 Modelar Base de Datos		A	P					R	
<b>2.5 Diseñar Prototipo de Interfaces</b>									
2.5.A01 Elaborar Prototipos				R	P				
2.5.A02 Verificar Prototipos con la Secretaria		P		R				A	
2.5.A03 Ejecutar correcciones				R	P				
2.5.A04 Obtención de conformidad de prototipos		P		R				A	
<b>3.0 CONSTRUCCION</b>									
3.1 Implementar Base de Datos en MySql	V							R	
3.2 Diseñar Interfaces del Sistema	V	A	R	P					
<b>3.3 Sub-Sistema de Control de Sacramentos</b>									
3.3.A01 Codificar Módulo de Feligrés			R	P					
3.3.A02 Ejecutar Pruebas del Módulo		A					R		P
3.3.A03 Elaborar Acta de Pruebas	V	A					R		
3.3.A04 Codificar Módulo de Sacramentos			R	P					
3.3.A05 Ejecutar Pruebas del Módulo		A					R		P
3.3.A06 Elaborar Acta de Pruebas	V	A					R		
3.3.A07 Codificar Módulo de Reportes de Sacramentos			R	P					

3.3.A08 Ejecutar Pruebas del Módulo			A		R	P
3.3.A09 Elaborar Acta de Pruebas	V	A			R	
<b>3.4 Sub-Sistema de Asignación de Personal</b>						
3.4.A01 Codificar Módulo de Personal			R	P		
3.4.A02 Ejecutar Pruebas del Módulo		A			R	P
3.4.A03 Elaborar Acta de Pruebas	V	A			R	
3.4.A04 Codificar Módulo de Comunidades			R	P		
3.4.A05 Ejecutar Pruebas del Módulo		A			R	P
3.4.A06 Elaborar Acta de Pruebas	V	A			R	
3.4.A07 Codificar Módulo de Reportes de Asignación			R	P		
3.4.A08 Ejecutar Pruebas del Módulo		A			R	P
3.4.A09 Elaborar Acta de Pruebas	V	A			R	
3.5 Codificar Módulo de Backups			R	P		
3.6 Ejecutar Pruebas del Módulo		A			R	P
3.7 Elaborar Acta de Pruebas	V	A			R	
3.8 Elaborar Manual Técnico	V	A	R	P		
3.9 Elaborar Manual de Usuario	V	A	R	P		
<b>4.0 TRANSICION</b>						
4.1 Ejecutar Pruebas Integrales	V	P			R	
4.2 Corrección de Errores Detectados			R	P		
4.3 Elaborar Acta de Pruebas	V	A			R	
4.4 Implantación del Sistema en la Parroquia	R					
4.5 Capacitación a los Usuarios	R	P				P
4.6 Entrega de Manuales	R	P				A
4.7 Elaborar Acta de Aceptación	V	A			R	
<b>5.0 GESTION DEL PROYECTO</b>						
<b>5.1 INICIO</b>						
<b>5.1.1 Elaborar Acta de Constitución</b>						
5.1.1.1 Reunión con el Párroco y la Secretaría	P	R	P			P
5.1.1.2 Elaborar Acta	A	R				
5.1.2 Identificar Interesados	V	R				
<b>5.2 PLANIFICACION</b>						
5.2.1 Elaborar Plan de Gestión de Py	A	R				
5.2.2 Elaborar Plan de Gestión de Cambios	A	R				
5.2.3 Elaborar Plan de Gestión de Requerimientos	A	R	P			
5.2.4 Elaborar Plan de Gestión de Alcance	A	R	P			
5.2.5 Elaborar EDT	A	V			R	
5.2.6 Elaborar Cronograma	A	V			R	
5.2.7 Elaborar Presupuesto	A	R				
5.2.8 Elaborar Plan de Gestión de Calidad del Py	A	R			P	
5.2.9 Elaborar Plan de Gestión de Comunicaciones	A	R	P			

5.2.10 Elaborar Plan de Gestión de Riesgos	A	R						
<b>5.2.11 Elaborar Plan de Gestión de RRHH</b>								
5.2.11.A01 Matriz RAM		R						
5.2.11.A02 Plan de Gestión de RRHH	A	R						
5.2.12 Elaborar Plan de Gestión de Adquisiciones	A	R	P			P		
<b>5.3 EJECUCION</b>								
5.3.1 Dirigir y Gestionar la ejecución del Py		R	P			P		
<b>    5.3.2 Aseguramiento de la Calidad</b>								
5.3.2.A01 Documentar cambios solicitados		R						
5.3.2.A02 Actualizar plan de gestión del Py		R	P					
5.3.3 Definir Acciones Correctivas recomendadas		R	P					
5.3.4 Realizar la distribución de la información	A	R	P					P
<b>5.4 SUPERVISION Y CONTROL</b>								
<b>    5.4.1 Supervisar y controlar el trabajo del Py</b>								
5.4.1.A01 Documentar las acciones correctivas recomendadas			R					
5.4.1.A02 Documentar las acciones preventivas recomendadas			R					
5.4.1.A03 Documentar cambios solicitados		R	P					
<b>    5.4.2 Control Integrado de Cambios</b>								
5.4.2.A01 Documentar solicitudes de cambio solicitadas		R						
5.4.2.A02 Documentar solicitudes de cambio rechazadas		R						
5.4.2.A03 Documentar acciones correctivas aprobadas		R						
5.4.2.A04 Documentar acciones preventivas aprobadas		R						
5.4.2.A05 Definir los Productos Entregables		R	P			P		
<b>    5.4.3 Verificación del Alcance</b>								
5.4.3.A01 Documentar los productos entregables aceptados			R					
5.4.3.A02 Documentar los cambios solicitados		R						
5.4.3.A03 Documentar las acciones correctivas recomendadas		R	P					
<b>    5.4.4 Control del Alcance</b>								
5.4.4.A01 Documentar los cambios solicitados		R						
5.4.4.A02 Documentar las acciones correctivas recomendadas		R	P					
<b>    5.4.5 Control del Cronograma</b>								
5.4.5.A01 Documentar los cambios solicitados		R				P		
5.4.5.A02 Documentar las acciones correctivas recomendadas		R	P			P		
<b>    5.4.6 Control de Costos</b>								
5.4.6.A01 Documentar la conclusión proyectada		R						

5.4.6.A02 Documentar los cambios solicitados		R					
5.4.6.A03 Documentar las acciones correctivas recomendadas		R	P				
<b>5.4.7 Control y Seguimiento de Riesgos</b>							
5.4.7.A01 Documentar riesgos surgidos		R					
5.4.7.A02 Documentar acciones realizadas		R					
<b>5.4.8 Control de Calidad</b>							
5.4.8.A01 Documentar la reparación de defectos recomendada		R				P	
5.4.8.A02 Documentar los productos entregables validados		R				P	
5.4.8.A03 Definir Acciones Preventivas recomendadas		R	P			P	
<b>5.5 CIERRE</b>							
<b>5.5.1 Cierre del Py</b>							
<b>5.5.1.1 Cierre Administrativo</b>							
<b>5.5.1.1.1 Lecciones Aprendidas</b>							
5.5.1.1.1.1 Documentar Lecciones Aprendidas	A	R	P			P	
<b>5.5.2 Cierre del Contrato</b>							
<b>5.5.2.1 Aceptación Final del Py</b>							
5.5.2.1.1 Entregar Producto Final	A	R				P	

Códigos de Responsabilidades: R = Responsable del Entregable A = Aprueba el Entregable P = Participa V = Revisa	Códigos de Roles: SP = Sponsor JP = Jefe de Proyecto AF = Analista Funcional PS = Programador Senior PJ = Programador Junior AC = Analista de Calidad DBA = Administrador de Base de Datos LU = Líder Usuario
-----------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 13 Matriz RAM

## Plan de Gestión del Personal

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	JC	LL	VD	04/07/2013	Original

### PLAN DE GESTIÓN DE RECURSOS HUMANOS

Nombre del Proyecto	Siglas del Proyecto
Implementación de un Sistema de Gestión para el Control de Sacramentos y Asignación de Personal	ISGCSAP

COMPONENTE	DESCRIPCIÓN		
Jefe del Proyecto:	Juan Castillo Hernández		
<b>ORGANIGRAMA DEL PROYECTO</b>			
Ver Organigrama del Proyecto (EDT) – versión 1.0			
<b>ROLES Y RESPONSABILIDAD</b>			
Ver Matriz de Asignaciones de Responsabilidades (RAM) – versión 1.0			
<b>Requerimiento de Personal</b>  JP = Jefe de Proyecto (D) AF = Analista Funcional (P) PS = Programador Senior (P) PJ = Programador Junior (P) AC = Analista de Calidad (V) DBA = Administrador de Base de Datos (V) LU = Líder Usuario (V)  (D): Dirige (P): Participa (V): Valida	<b>Descripción de Roles</b>  - Planificación y Aseguramiento de la Calidad. - Aseguramiento de la Calidad. - Control de Calidad - Control de Calidad - Aseguramiento de la Calidad. - Aseguramiento de la Calidad. - Aseguramiento de la Calidad		
<b>OBJETIVO DEL PLAN</b>	Asegurar el reclutamiento de personal idóneo para el Proyecto y que por tanto cumpla con el perfil que se detalla más adelante.		
<b>REQUERIMIENTO DE PERSONAL</b>			
Será al inicio de las operaciones del Proyecto. Los cargos serán los siguientes: Jefe de Proyecto, Analista Funcional, Programador Senior, Programador Junior, Analista de Calidad, Administrador de Base de datos.			
<b>CRITERIOS DE LIBERACIÓN DEL PERSONAL DEL PROYECTO:</b>			
ROL	CRITERIO DE LIBERACIÓN	¿CÓMO?	DESTINO DE ASIGNACIÓN
Sponsor	Al término del proyecto.		
Jefe de proyecto	Al término del proyecto.	Comunicación del Sponsor	
Analista Funcional	Al término del proyecto.	Comunicación del Jefe de Proyecto	

Programador Senior	Al terminar sus entregables	Comunicación del Jefe de Proyecto	
Programador Junior	Al terminar sus entregables	Comunicación del Jefe de Proyecto	
Analista de Calidad	Al terminar sus entregables	Comunicación del Jefe de Proyecto	
Administrador de Base de Datos	Al terminar sus entregables	Comunicación del Jefe de Proyecto	
Líder Usuario	Al término del proyecto.	Comunicación del Jefe de Proyecto	
<b>CAPACITACIÓN, ENTRENAMIENTO, MENTORING REQUERIDO</b>			
1. Siempre se deben aprovechar los proyectos para que el Project Manager más experimentado haga mentoring a los menos experimentados, en este caso el Sponsor hará mentoring al Jefe de Proyecto para ayudarlo a desarrollar sus habilidades de gestión de proyectos; asimismo el Jefe de Proyecto hará mentoring a los especialistas que están bajo su cargo.			
<b>SISTEMA DE RECONOCIMIENTOS Y RECOMPENSA</b>			
<b>CUMPLIMIENTO DE REGULACIONES, PACTOS Y POLÍTICAS</b>			
1. Solo se debe adquirir personal que cuente con experiencia en desarrollo de sistemas.			
<b>REQUERIMIENTO DE SEGURIDAD</b>			
1. La enfermedad del jefe del proyecto o algún miembro del equipo del proyecto generará retraso en el cronograma del proyecto, debido a que las tareas que se le fueron asignados a ese personal no podrán ser realizadas mientras esté enfermo, por tanto se fija como requerimiento de seguridad, tener disponible algún reemplazo en caso que el principal no pueda continuar con las tareas asignadas. 2. La documentación física generada del proyecto, en una mala ubicación puede causar su pérdida o deterioro por factores ambientales o humanos intencionales, por tanto se fija como requerimiento de seguridad el guardado de la documentación en un lugar de difícil acceso a personas ajenas al proyecto y su digitalización y subida a una carpeta en la nube.			
<b>APROBACIONES</b>			
 Juan Guillermo Castillo Hernández Gerente del Proyecto 04/07/2013		 Vicente Díez Varona Pároco Patrocinador del Py 04/07/2013	

## E. Comunicaciones

### Directorio de Stakeholders

NOMBRE	ROL EN EL PROYECTO	LOCALIZACIÓN	NÚMEROS DE CONTACTO	CORREO ELECTRÓNICO
Vicente Diez Varona	Sponsor	PISCO	05653-2148	
Luisa Cajo Chacalizaza	Líder de usuario	PISCO	05653-2148	<a href="mailto:Luisa_es_14@hotmail.com">Luisa_es_14@hotmail.com</a>
Juan Castillo Hernández	Jefe de Proyecto	PISCO	956447724 *693387	<a href="mailto:Juancas.hernandez92@gmail.com">Juancas.hernandez92@gmail.com</a>
Luis Loo Parián	Asesor	ICA	982369996 956835614	<a href="mailto:lloop@grupoalkas.org">lloop@grupoalkas.org</a> <a href="mailto:lloop@live.com">lloop@live.com</a> <a href="mailto:loo.parian.luis@gmail.com">loo.parian.luis@gmail.com</a>
Hugo Núñez Román	Asesor	PISCO	989073874	<a href="mailto:h_nunez@uap.edu.pe">h_nunez@uap.edu.pe</a>

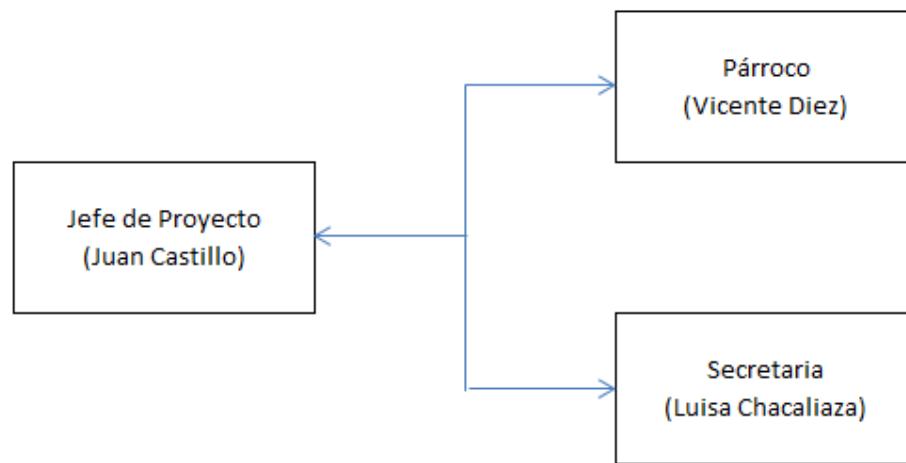
Tabla 14 Directorio de Stakeholders

## **Plan de Comunicaciones**

### **Definición del Esquema de Comunicaciones**

Es importante planear la manera cómo comunicar los avances, entregables y manejo de requerimientos entre el equipo de Proyecto y los Usuarios. Este plan deberá permitir que todos los integrantes del proyecto tengan definido correctamente qué información necesita cada involucrado en el proyecto en el momento oportuno.

En el siguiente diagrama se sintetiza el esquema de comunicaciones a seguir en el proyecto:



### **Tipos de Comunicación**

Entre Jefe de Proyecto y Párroco: vía reunión con los involucrados.

Entre Jefe de Proyecto y Secretaria: vía reunión o vía telefónica.

### **Esquema de Comunicación de Documentos**

A continuación se presenta el esquema de comunicación de los principales Documentos del proyecto.

<b>Documento</b>	<b>Tipo</b>	<b>Usuario Destino</b>	<b>Medio de Entrega</b>	<b>Formato</b>	<b>Frecuencia</b>	<b>Responsable</b>
Status Report de Avances	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Mitad y Fines de Mes	Juan Castillo
Acta de Reunión	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Hasta 2 días luego de Reunión	Juan Castillo
Documento de Análisis	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Hito del Plan	Juan Castillo
Documento de Diseño	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Hito del Plan	Juan Castillo
Acta de Aceptación de Entregable de Documento de Análisis	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Luego de Entregado el Documento de Análisis	Juan Castillo
Acta de Aceptación de Entregable de Documento de Diseño	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Luego de Entregado el Documento de Diseño	Juan Castillo
Acta de Aceptación del Proyecto	Obligatorio	Vicente Diez	Plantilla, reunión	Word	Luego de Entregados los Entregables del Proyecto	Juan Castillo
Cambios Al Alcance	Informativo	Todos los involucrados	Plantilla, reunión	Word	Antes de una reunión por cambio al alcance	Juan Castillo
Documento de Impacto del Cambio	Obligatorio	Todos los involucrados	Plantilla, reunión	Word	Después de una reunión por cambio al alcance	Juan Castillo

al Alcance						
Información del Proyecto	Informativo	Todos los involucrados	reunión	Word	Quincenal	Juan Castillo
Logros del Proyecto	Marketing	Todos los involucrados	reunión	Word	Quincenal	Juan Castillo

### **Estándares para las Reuniones**

- En toda reunión debe estar definida una agenda.
- La agenda y el material de la reunión deben estar preparados antes de la ejecución de la reunión.
- El acta de reunión debe ser enviada como máximo a los dos (2) días de ejecutada la reunión.
- Los participantes deben prepararse adecuadamente revisando la agenda y el material preparado.
- Deben empezar y terminar según lo programado.

### **Estándares para los Archivos**

- Para los Entregables Documento de Análisis y Documento de Diseño, se usará el formato Word.
- Para presentaciones, se usará el formato Power Point.
- Para hojas de cálculo, se usará el formato Excel.
- Para el calendario del proyecto, se usará el formato MS Project.
- Para archivos comprimidos se usará el formato 7Zip.

### **Almacenamiento de Documentación del Proyecto**

Toda la documentación del proyecto será almacenada en un repositorio en Google Drive. Se creará una carpeta llamada “Proyecto” que será administrado por el Gerente de Proyecto (con permisos de lectura y escritura). La información será almacenada con la siguiente estructura:

- 1.1 Estado del Proyecto
  - 1.1.1 Organización del Equipo y Asignación de responsabilidades
  - 1.1.2 Cronograma del Proyecto
  - 1.1.3 Log de Control de Cambios al Alcance
  - 1.1.4 Log de Control de Cambios al Cronograma
  - 1.1.5 Log de Control de Cambios al Costo
- 1.2 Correspondencia al Líder del Proyecto
  - 1.2.1 Actas de Reunión
  - 1.2.2 Reportes de Avance
  - 1.2.3 Verificación de Trabajo Realizado en los Hitos
  - 1.2.4 Actas de Aceptación de Entregables y Cierre de Proyecto
- 1.3 Correspondencia Interna
  - 1.3.1 Documentación de Respaldo
- 1.4 Entregables Principales
  - 1.4.1 Documento de Análisis
    - 1.4.1.1 Documento de Análisis Versión 1.0 completado al hito.
  - 1.4.2 Documento de Diseño
    - 1.4.2.1 Documento de Diseño Versión 1.0 completado al hito.

## Plantilla de Acta de Reunión

### Datos de Reunión

Fecha y Hora :	
Lugar :	
Organizador :	

### Participantes

Nombre	Área	Nemónico (siglas del participante, así se referenciará a los participantes en el acta)

### Temas Tratados

#### Nombre del Tema o Requerimiento :

Tema	Responsable
<b>“Nombre del Tema”</b>	
Tema 1	Responsable 1
Tema n	Responsable n
<b>“Acuerdos”</b>	
Acuerdo 1	Responsable 1
Acuerdo n	Responsable n

Concluida la presente reunión a las HH:MM horas, se levanta la presente acta, firmándose en duplicado en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

*Por el usuario*

---

Líder usuario

---

Usuario

*Por el Equipo de Proyecto*

---

Gerente de Proyecto

---

Analista Proyecto

## **Plantilla de Requerimientos de Cambio al Alcance / Nueva Funcionalidad**

### **Datos del Solicitante**

Código de Petición :		
Fecha de Solicitud:		
Nombre Solicitante		
Área Promotora		
Cargo solicitante	Teléfono	E-Mail

### **Datos de la Petición de Cambio / Nueva Funcionalidad**

Título :	"Título "
Prioridad:	(Máxima, Media, Baja)
Descripción Simplificada de la Petición	
Descripción del Motivo	
Documentos Adjuntos	

### **Del Cambio o Nuevo Requerimiento Propuesto**

Objetivo :	
Alcance:	

### **Mejoras o Beneficios Esperados**

(Identificar, definir, clasificar e interpretar los objetivos traducidos en beneficios esperados por el usuario, con el que deberá contar el requerimiento propuesto, que espera el usuario como solución a sus necesidades, plasmándola en mejoras o beneficios descritos clara y concretamente, diferenciados y numerados, asegurando que su descripción no dé lugar a ambigüedades)

Ítem :	Descripción de la Mejora o Beneficio	Importancia (alta, media, baja)

## **Plantilla de Cambio al Cronograma**

### **Datos del Solicitante**

Código de Petición :		
Fecha de Solicitud:		
Nombre Solicitante		
Área Promotora		
Cargo solicitante	Teléfono	E-Mail

### **Datos de la Petición de Cambio al Cronograma**

Título :	"Título "
Prioridad:	(Máxima, Media, Baja)
Descripción Simplificada de la Petición	
Descripción del Motivo	
Documentos Adjuntos	

### **Del Cambio al Cronograma**

Objetivo :	
Alcance:	

### **Riesgos involucrados de no implementar el Cambio al Cronograma** (Identificar y definir los riesgos de no implementarse el cambio al cronograma)

Ítem :	Descripción del Riesgo	Responsable del Riesgo	Impacto en el Proyecto (A, M, B)

## Plantilla de Acta de Cierre de Proyecto

### Datos de Lugar y Fecha

Fecha / Hora:	
Lugar	
Referencia	ACTA DE CIERRE DE PROYECTO

### Participantes

Nombre	Área (Usuaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco

### Objetivo del Documento

El presente documento tiene como objetivo dar la conformidad de cada uno de los puntos ofrecidos para la solución. Asimismo se indican las observaciones realizadas por el usuario.

### Observaciones

A continuación se detalla cada una de las funcionalidades ofrecidas en el documento.

Ítem :	Funcionalidad	Estado	Observación
1	DOCUMENTO DE ANÁLISIS		
2	DOCUMENTO DE DISEÑO		

### Conclusiones

Se definen como totalmente aprobadas los entregables verificados por los usuarios responsables

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

Nombre del Participante	Firma
Juan Castillo	
Vicente Diez	

## Plantilla de Reporte de Status de Avance

Fecha / Hora:	
Lugar	
Elaborado por	Juan Castillo
Gerencia/Área	GERENCIA PROYECTO
Período	Del _____ Al _____
Referencia	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO

## **Plantilla de Acta de Aceptación de Documento de Análisis**

### **Datos de Lugar y Fecha**

Fecha / Hora:	
Lugar	
Referencia	PLANTILLA DE ACTA DE ACEPTACIÓN DE DOCUMENTO DE ANÁLISIS

### **Participantes**

Nombre	Área (Usaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco

### **Objetivo del Documento**

El presente documento tiene como objetivo dar la conformidad de cada uno de los puntos ofrecidos para la solución. Asimismo se indican las observaciones realizadas por el usuario.

### **Observaciones**

A continuación se detalla cada una de las funcionalidades ofrecidas en el documento.

Ítem :	Nombre	Descripción	Estado	Observación

### **Conclusiones**

Se definen como totalmente aprobados los ítems del Documento de Análisis por los usuarios responsables

### **Aceptación Final**

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

Nombre del Participante	Firma
Juan Castillo	
Vicente Diez	

## **Plantilla de Acta de Aceptación de Documento de Diseño**

### **Datos de Lugar y Fecha**

Fecha / Hora:	
Lugar	
Referencia	PLANTILLA DE ACTA DE ACEPTACIÓN DE DOCUMENTO DE DISEÑO

### **Participantes**

Nombre	Área (Usaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco

### **Objetivo del Documento**

El presente documento tiene como objetivo dar la conformidad de cada uno de los puntos ofrecidos para la solución. Asimismo se indican las observaciones realizadas por el usuario.

### **Observaciones**

A continuación se detalla cada uno de los ítems ofrecidos en el documento.

Ítem :	Nombre	Descripción	Estado	Observación

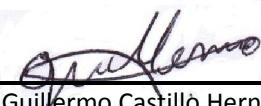
### **Conclusiones**

Se definen como totalmente aprobados los ítems del Documento de Diseño por los usuarios responsables

### **Aceptación Final**

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

Nombre del Participante	Firma
Juan Castillo	
Vicente Diez	

APROBACIONES	
 Juan Guillermo Castilló Hernández Gerente del Proyecto 06/07/2013	 Vicente Diez Párroco Patrocinador del Py 06/07/2013

## F. Riesgos

### Listado de Riesgos

ID	Descripción del Riesgo	Tipo de Riesgo
R01	Requisitos poco claros	Riesgo del Producto
R02	Falta de Experiencia en tareas de planificación	Riesgo del Proyecto
R03	Falta de Experiencia con las herramientas utilizadas	Riesgo del Producto/Proyecto
R04	Diseño Erróneo	Riesgo del Producto
R05	Falta de un Experto	Riesgo del Proyecto
R06	Pérdida de documentación y/o otros artefactos	Riesgo del Proyecto
R07	Control de Calidad inadecuado	Riesgo del Proyecto
R08	Estimación de costos fuera del alcance de la realidad	Riesgo del Proyecto
R09	Falta de seguimiento permanente de tareas y actividades	Riesgo del Proyecto
R10	Excesivo tiempo en corregir errores	Riesgo del Producto

Tabla 15 Listado de Riesgos

## Identificación, Estimación y Priorización de Riesgos

Id.	Descripción del Riesgo	Tipo Riesgo	Probab. Ocurrencia	Nivel de Impacto	Evaluación del Riesgo	Acciones de Prevención	Acción de Corrección
R01	Cambios en los Requisitos	Producto	20	4	0.8	Realización de reuniones con el cliente para la aclaración de requisitos.	Se revisarán los requisitos afectados, así como toda la documentación y código derivado de los mismos hasta el punto de aparición del cambio.
R02	Falta de Experiencia en tareas de planificación	Proyecto	50	2	1	El uso de RUP como proceso de desarrollo. Realización de reuniones entre los miembros del proyecto para la evaluación de la marcha del proyecto y consultas al tutor.	Se observarán las diferencias entre la planificación de cada iteración y el informe de seguimiento de su ejecución, para tratar de detectar y corregir errores de planificación en las iteraciones posteriores.
R03	Falta de Experiencia con las herramientas utilizadas	Producto /Proyecto	50	2	1	Una parte del tiempo de desarrollo del proyecto se destinará al aprendizaje de las herramientas de documentación e implementación.	Si se produce un retraso por parte de un miembro del equipo, los demás miembros tratarán de ayudar a superarlo. Consultar a fuentes externas En último lugar se haría una redistribución de tareas.

R04	Diseño Erróneo	Producto	40	3	1.2	Durante la fase de Elaboración se desarrollará en paralelo un prototipo conteniendo la arquitectura del sistema para comprobar la validez de la misma.	Se revisará y modificará la documentación de diseño afectada. La planificación se reajustará si fuera necesario.
R05	Falta de un Experto	Proyecto	80	1	0.8	Aprendizaje continuo durante todo el proyecto	Las dudas que no se sepan resolver se trasladarán al tutor y a foros especializados.
R06	Pérdida de documentación y/o otros artefactos	Proyecto	40	4	1.6	Se realizarán copias de seguridad en los dispositivos externos a disposición, así como también el alojamiento en la nube.	Actualizar con la última copia disponible
R07	Control de Calidad inadecuado	Proyecto	60	4	2.4	Comprobar si se están siguiendo los estándares fijados en la planificación así como también los requerimientos establecidos.	Se estudiará posibles soluciones teniendo en cuenta el tiempo que se tiene para el proyecto. Si es necesario se harán reajustes a los documentos involucrados.

R08	Mala estimación de costos	Proyecto	50	3	1.5	Realización de varias estimaciones con metodologías diferentes	Redimensionar el proyecto conforme se ejecuta
R09	Falta de seguimiento de tareas	Proyecto	50	3	1.5	Llevar al día una revisión del estado del proyecto para anotar los posibles atrasos y poder así tomar medidas en el instante	Recandelerización de las tareas, charla con el equipo de desarrollo en caso de detectarse malas prácticas.
R10	Excesivo tiempo en corregir errores	Producto	40	3	1.2	Uso de un estándar de programación y/u ordenamiento del mismo para ubicar fácilmente trozos de código que ejecuten cada acción.	Se acudirá a copias de seguridad antes de codificar la parte que genera errores, de este modo si no puede ubicarse se puede codificar de nuevo teniendo como modelo el código con error e identificar estos errores fácilmente. División del código y prueba de cada trozo por separado para identificar en qué parte se encuentra el error.

Tabla 16 Matriz de Riesgos

## Documento de Análisis de Riesgos del Proyecto

ID	Análisis del Riesgo
R01	<p><b>Magnitud</b>  Variable según la fase de aparición:</p> <ul style="list-style-type: none"> <li>▪ Inicio: baja.</li> <li>▪ Elaboración: media.</li> <li>▪ Construcción: alta.</li> <li>▪ Transición: muy alta</li> </ul> <p><b>Descripción</b>  Los requisitos representan la idea que tiene el cliente sobre la aplicación, sobre ellos se construyen los casos de uso y dichos casos de uso guían el desarrollo del proyecto. Una mala o insuficiente recolección de los mismos afecta a la calidad de todo el proyecto.</p> <p><b>Impacto</b>  La incorporación o modificación de requisitos durante el desarrollo requerirá realizar cambios sobre gran parte de la documentación del producto elaborada con anterioridad al momento del cambio. Estas modificaciones serán menos costosas durante las dos primeras fases del proyecto, pero pueden suponer trastornos importantes durante las fases de Construcción y Transición, pues no sólo cambiaría la documentación sino también el código fuente y los ejecutables.</p> <p><b>Indicadores</b>  Al realizar la consulta al cliente, no sabe indicar con propiedad cuales son los servicios que espera obtener de la aplicación.</p>
R02	<p><b>Magnitud</b>  Media.</p> <p><b>Descripción</b>  El responsable tiene poca experiencia en el desarrollo de software siguiendo una estructura de tareas y fechas preestablecido.</p> <p><b>Impacto</b>  La planificación guía todo el desarrollo del proyecto. Un error en la misma puede incidir directamente en sus resultados. No obstante, la división en iteraciones reduce el posible impacto de los errores, permitiendo que estos puedan ser corregidos o absorbidos en iteraciones posteriores a la de su aparición.</p> <p><b>Indicadores</b>  Diferencias entre el desarrollo real del proyecto y la planificación estimada.</p>
R03	<p><b>Magnitud</b>  Variable según la fase de aparición:</p> <ul style="list-style-type: none"> <li>▪ Inicio: baja.</li> <li>▪ Elaboración: media.</li> <li>▪ Construcción: alta.</li> <li>▪ Transición: alta.</li> </ul> <p><b>Descripción</b>  El responsable tiene dificultades a la hora de realizar sus objetivos (tanto de documentación como de implementación) por su inexperiencia con las herramientas disponibles para el mismo.</p> <p><b>Impacto</b>  Puede suponer retrasos.</p>

	<p><b>Indicadores</b> No procede.</p>
R04	<p><b>Magnitud</b> Baja en Elaboración, alta en Construcción.</p> <p><b>Descripción</b> El diseño del sistema resulta inadecuado. Al realizar actividades de implementación puede encontrarse que el diseño carece del suficiente nivel de detalle o está mal enfocado, bien por la naturaleza del problema, o bien por restricciones de uso impuestas por tecnologías de terceros.</p> <p><b>Impacto</b> Puede introducir retrasos en el proyecto ante la necesidad de volver a considerar el diseño trazado. Requiere la actualización o modificación de los artefactos de diseño.</p> <p><b>Indicadores</b> La arquitectura no cumple las expectativas. Se complica la implementación.</p>
R05	<p><b>Magnitud</b> Media.</p> <p><b>Descripción</b> No hay un experto del dominio en el equipo de consultoría o en el desarrollo, al que poder consultar.</p> <p><b>Impacto</b> Puede suponer retrasos.</p> <p><b>Indicadores</b> No procede</p>
R06	<p><b>Magnitud</b> Alta.</p> <p><b>Descripción</b> Por causas varias se pierde parte o el total de la documentación así como la posibilidad de perder parte o el total de otros artefactos, como pueden ser: parte de la implementación o ficheros de planificación.</p> <p><b>Impacto</b> Variable, puede suponer una catástrofe, o un simple retraso.</p> <p><b>Indicadores</b> Ninguno.</p>
R07	<p><b>Magnitud</b> Alta.</p> <p><b>Descripción</b> Control de calidad escaso o inapropiado, por falta de experiencia y/o conocimientos.</p> <p><b>Impacto</b> Si no se lleva un control de calidad correcto durante la construcción del producto, esto puede ocasionar que la calidad de este sea diferente al que se planeó, lo cual puede ocasionar que el cliente rechace el producto y este tenga que ser rediseñado, utilizando tiempo que le corresponde a otras actividades.</p>

	<p><b>Indicadores</b> Durante el desarrollo del producto no se le da importancia al control de calidad.</p>
R08	<p><b>Magnitud</b> Media</p> <p><b>Descripción</b> Se sobreestiman o subestiman los costos involucrados con el desarrollo del producto de software</p> <p><b>Impacto</b> Puede generar que el equipo entre en períodos de sobrecarga de trabajo o períodos de ausencia del mismo, lo que a su vez puede conllevar a un deterioro en la calidad</p> <p><b>Indicadores</b> El equipo trabaja más o menos horas de las inicialmente programadas, se presentan quejas a jefe del Proyecto o Pedidos de redimensionamiento</p>
R09	<p><b>Magnitud</b> Media</p> <p><b>Descripción</b> No se realiza un seguimiento de las tareas planificadas, lo que puede ocasionar que algunas de ellas sean dejadas para última instancia, con la consecuente baja en su calidad</p> <p><b>Impacto</b> Sobrecarga de trabajo en los días previos a la entrega de un entregable, pobre calidad de los entregables, se obvian detalles importantes.</p> <p><b>Indicadores</b> No se revisa la documentación de planificación y si se revisa no se le da la debida importancia al seguimiento de actividades planeadas ni al resultado.</p>
R10	<p><b>Magnitud</b> Alta</p> <p><b>Descripción</b> La corrección de los errores encontrados en las pruebas lleva más tiempo del planeado.</p> <p><b>Impacto</b> Puede generar retrasos así como también que se vuelvan a desarrollar módulos que ya se encontraban terminados.</p> <p><b>Indicadores</b> El cliente y/o el jefe de proyecto anuncian al equipo el cambio de tecnología.</p>

Tabla 17 Cuadro de Análisis de Riesgos

## **G. Adquisiciones**

### **Infraestructura, Equipos, Materiales y Accesorios**

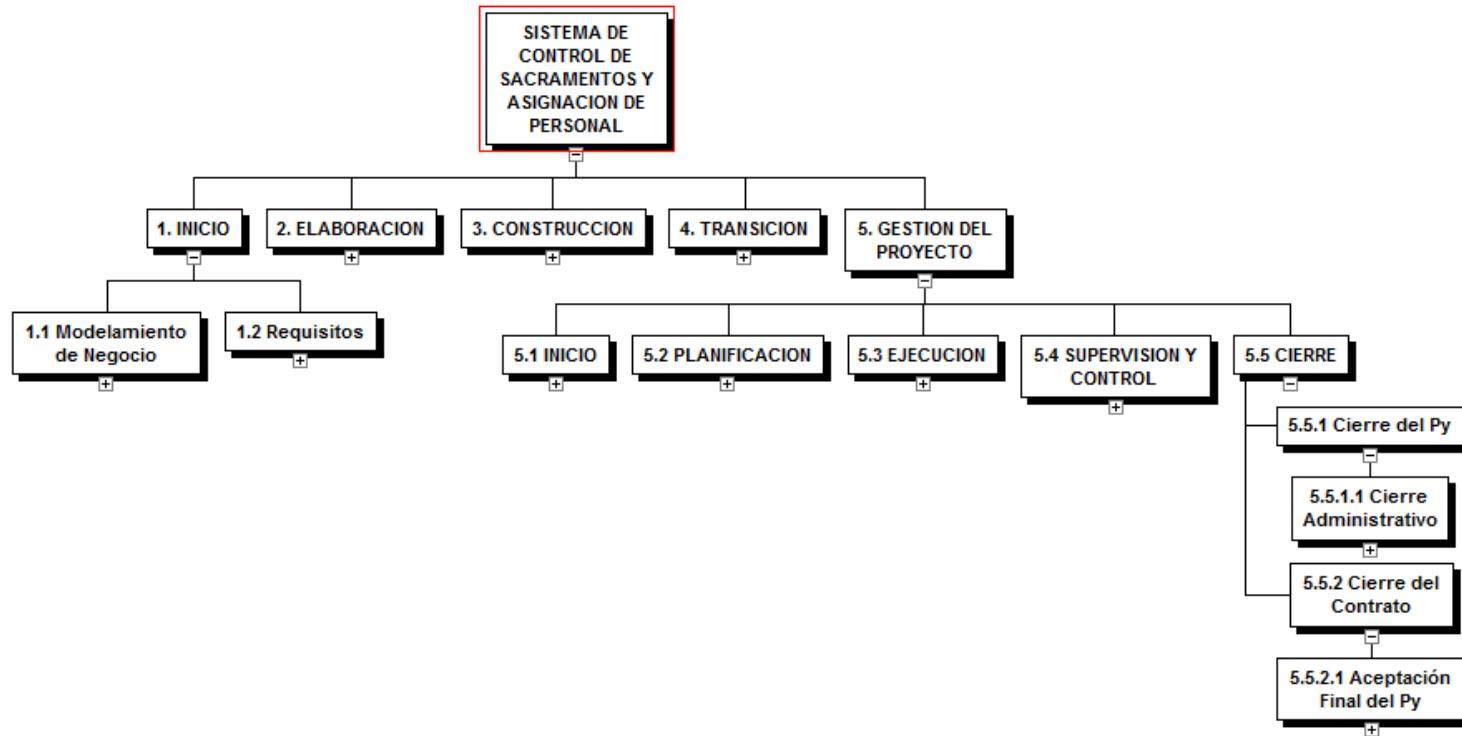
#### **Materiales**

- 2 Millares de Hojas Bond A-4 75 gramos
- 3 Pomo de Tinta Negra
- 1 Pomo de Tinta Azul
- 1 Pomo de Tinta Amarillo
- 1 Pomo de Tinta Rojo
- 1 Lapicero PILOT Tinta Negra
- 1 Lapicero PILOT Tinta Azul
- 5 Folder Manila A-4
- 2 Archivadores Artesco
- 1 Perforadora de Papel
- 20 Micas Portapapeles

#### **Equipos**

- 1 PC Escritorio As-Rock NS-64 300 GB, 2 GB RAM
- 1 Impresora Multifuncional Canon MP 495
- 1 Memoria USB 8 GB

## Plan de Adquisiciones



## Matriz de Adquisiciones

NOMBRE DEL PROYECTO: SISTEMA DE CONTROL DE SACRAMENTOS Y ASGINACION DE PERSONAL											
CTO O SERVICIO A ADQUIRIR	CÓDIGO DE ELEMENTO WBS	TIPO DE CONTRATO	DOCUMENTOS DE LA ADQUISICIÓN	REQUERIMIENTO DE ESTIMACIONES INDEPENDIENTES	ÁREA/ROL/PERSONA RESPONSABLE DE LA COMPRA	MANEJO DE MÚLTIPLES PROVEEDORES	PROVEEDORES PRE-CALIFICADOS	CRONOGRAMA			
								Planif. ADQUIS	Efectuar ADQUIS	Admin. ADQUIS	Cerrar ADQUI S
								Del al	Del al	Del al	Del al
Millar de Hojas Bond A-4 75 gramos		Precio Fijo	-Boleta		Juan Castillo		Multiservicios "Nato's".	Plan de Adquisiciones. 05/07/2013	05/07/2013	10/07/2013	21/07/2013
Pomo de Tinta Negra		Precio Fijo	-Boleta		Juan Castillo		OK.com	Plan de Adquisiciones. 05/07/2013	05/07/2013	10/07/2013	21/07/2013
Pomo de Tinta Azul		Precio Fijo	-Boleta		Juan Castillo		OK.com	Plan de Adquisiciones. 05/07/2013	05/07/2013	10/07/2013	21/07/2013
Pomo de Tinta Amarillo		Precio Fijo	-Boleta		Juan Castillo		OK.com	Plan de Adquisiciones. 05/07/2013	05/07/2013	10/07/2013	21/07/2013
Pomo de Tinta Rojo		Precio Fijo	-Boleta		Juan Castillo		OK.com	Plan de Adquisiciones. 05/07/2013	05/07/2013	10/07/2013	21/07/2013
Lapicero PILOT Tinta Negra		Precio Fijo	-Boleta		Juan Castillo		Multiservicios "Nato's".	Plan de Adquisiciones. 05/07/2013	05/07/2013	11/07/2013	12/07/2013
Lapicero PILOT		Precio Fijo	-Boleta		Juan Castillo		Multiservicios "Nato's".	Plan de Adquisicio	05/07/20	11/07/20	12/07/2013

Tinta Azul							nes. 05/07/2013	13			
Folder Manila A-4		Precio Fijo	-Boleta		Juan Castillo		Multiservici os "Nato's".	Plan de Adquisicio nes. 05/07/2013	05/07/20 13	11/07/20 13	12/07/2 013
PC Escritorio As-Rock NS-64 300 GB, 2 GB RAM		Precio Fijo	-Boleta		Juan Castillo		Tiendas Efe	Plan de Adquisicio nes. 05/07/2013	06/07/20 13	08/07/20 13	13/07/2 013
Impresora Multifuncio nal Canon MP 495		Precio Fijo	-Boleta		Juan Castillo		Tiendas Efe	Plan de Adquisicio nes. 05/07/2013	06/07/20 13	06/07/20 13	13/07/2 013
Memoria USB 8 GB		Precio Fijo	-Boleta		Juan Castillo		OK.com	Plan de Adquisicio nes. 05/07/2013	06/07/20 13	06/07/20 13	13/07/2 013
Archivador es		Precio Fijo	-Boleta		Juan Castillo		Librería San Francisco 197	Plan de Adquisicio nes. 05/07/2013	10/07/20 13	11/07/20 13	12/07/2 013
Perforador a de papel		Precio Fijo	-Boleta		Juan Castillo		Librería San Francisco 197	Plan de Adquisicio nes. 05/07/2013	10/07/20 13	11/07/20 13	12/07/2 013

<b>ADQ01</b>	<b>Millar de Hojas Bond A-4 75 gramos</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Multiservicios "Nato's"		
<b>Descripción:</b> Millar de Hojas Bond A-4 de 75 gramos.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>▪ 2 Millar Hojas BOND.....S/. 48.00</li> </ul>		

<b>ADQ02</b>	<b>Pomo de Tinta Negra</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: OK.COM		
<b>Descripción:</b> Pomo de tinta negra JetLife 100 ML.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 3 Pomo de Tinta Negra..... S/.36.00</li> </ul>		

<b>ADQ03</b>	<b>Pomo de Tinta Azul</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: OK.COM		
<b>Descripción:</b> Pomo de tinta azul JetLife 100 ML.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Pomo de Tinta Azul..... S/.12.00</li> </ul>		

<b>ADQ04</b>	<b>Pomo de Tinta Amarillo</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: OK.COM		
<b>Descripción:</b> Pomo de tinta amarillo JetLife 100 ML.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Pomo de Tinta Amarillo..... S/.12.00</li> </ul>		

<b>ADQ05</b>	<b>Pomo de Tinta Roja</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: OK.COM		
<b>Descripción:</b> Pomo de tinta rojo JetLife 100 ML.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Pomo de Tinta Roja..... S/.12.00</li> </ul>		

<b>ADQ06</b>	<b>Lapicero PILOT Tinta Negra</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Multiservicios "Nato's"		
<b>Descripción:</b> Lapicero de tinta negra marca PILOT.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• Lapicero PILOT Tinta Negra..... S/.3.00</li> </ul>		

<b>ADQ07</b>	<b>Lapicero PILOT Tinta Azul</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Multiservicios “Nato’s”		
<b>Descripción:</b> Lapicero de tinta azul marca PILOT		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Lapicero PILOT Tinta Azul..... S/.3.00</li> </ul>		

<b>ADQ08</b>	<b>Folder Manila A-4</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Multiservicios “Nato’s”		
<b>Descripción:</b> Folder manila simple tamaño A-4.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 5 Folder Manila A-4..... S/.5.00</li> </ul>		

<b>ADQ09</b>	<b>PC Escritorio As-Rock NS-64 300 GB, 2 GB RAM</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Tiendas EFE		
<b>Descripción:</b> Computadora de escritorio: <ul style="list-style-type: none"> <li>• Placa: As-Rock N-64</li> <li>• Disco Duro: LG 300 GB</li> <li>• RAM: LG 2 GB</li> <li>• Monitor LG 21'</li> <li>• Teclado Genius</li> <li>• Mouse Genius</li> <li>• Parlantes Genius</li> </ul>		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 PC Escritorio..... S/.2,000.00</li> </ul>		

<b>ADQ10</b>	<b>Impresora Multifuncional Canon MP 495</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Tiendas EFE		
<b>Descripción:</b> Impresora multifuncional marca CANON. Impresora + Scanner.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Impresora Multifuncional..... S/.320.00</li> </ul>		

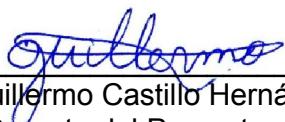
<b>ADQ11</b>	<b>Memoria USB 8 GB</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: OK.COM		
<b>Descripción:</b> Memoria USB marca Kingston de 8 GB		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Memoria USB 8GB..... S/.42.00</li> </ul>		

<b>ADQ12</b>	<b>Archivadores</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Librería San Francisco 197.		
<b>Descripción:</b> Archivadores Artesco.		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 2 Archivadores Artesco..... S/.32.00</li> </ul>		

<b>ADQ13</b>	<b>Perforadora de Papel</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Librería San Francisco 197.		
<b>Descripción:</b> Perforadora de Papel		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 1 Perforadora de Papel..... S/. 9.00</li> </ul>		

<b>ADQ14</b>	<b>Micas Portapapeles</b>	
<b>Nombre del proyecto: Sistema de Control de Sacramentos y Asignación de Personal</b>		
<b>Código de proyecto:</b> Proy-001	<b>Tipo de proyecto:</b> Software	<b>Cliente:</b> Parroquia San Clemente de Pisco
<b>Elaborado por:</b> Juan Castillo		<b>Aprobado por:</b> Vicente Diez
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor: Librería San Francisco 197.		
<b>Descripción:</b> Micas Portapapeles		
<b>Parámetros de Costo:</b> <ul style="list-style-type: none"> <li>• 20 Micas Portapapeles..... S/. 10.00</li> </ul>		

**Aprobaciones**

  
Juan Guillermo Castillo Hernández  
Gerente del Proyecto  
05/07/2013

  
Vicente Díez Varona  
Párroco  
Patrocinador del Py  
05/07/2013

### 3.2. Ingeniería del proyecto

#### 3.2.1. Modelamiento de requerimientos

Involucrado	Cod. Req	Descripción del requerimiento	Categoría	Prioridad	Criterio de aceptación
Secretaria	Req-001	Actualizar los datos de los países. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de países, durante las pruebas de Control de Calidad
Secretaria	Req-002	Actualizar los datos de las ciudades o departamentos. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de las ciudades, durante las pruebas de Control de Calidad
Secretaria	Req-003	Actualizar los datos de las provincias del Perú. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de las provincias, durante las pruebas de Control de Calidad
Secretaria	Req-004	Para los países, ciudades o departamento y provincias, sólo se podrán eliminar los nuevos registros que se ingresen en el sistema.	Req. Funcional	Media	Impedir la eliminación de los registros por defecto de los países, ciudades y provincias.
Secretaria	Req-005	Actualizar los datos de los templos: Nombre, tipo, lugar. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de los templos, durante

		Cancelar.			las pruebas de Control de Calidad
Secretaria	Req-006	Actualizar los datos del feligrés. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de los feligreses, durante las pruebas de Control de Calidad
Secretaria	Req-007	Posibilidad de registrar a un personal como feligrés y viceversa.	Req. Funcional	Media	Casilla de verificación en la interfaz de feligrés si se tiene los datos del personal. Casilla de verificación en la interfaz del personal si se tiene los datos del feligrés.
Secretaria	Req-008	Actualizar los datos del Bautismo: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Natural del Padre, Nombres de la Madre, Natural de la Madre, Lugar de nacimiento, Fecha de nacimiento, Fecha de Bautismo, Nombre del Padrino, Nombre de la Madrina, Ministro o Celebrante, Nota Marginal. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de bautismos, durante las pruebas de Control de Calidad
Secretaria	Req-009	Actualizar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios,	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz

		Nº de registros por folio, fecha de apertura, fecha de cierre, estado, cantidad de registros. Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reaperturar Libro, Buscar y Cancelar.			de los libros sacramentales, durante las pruebas de Control de Calidad
Secretaria	Req-010	Permitir la reapertura de un libro sacramental si el total de registros no es igual a la cantidad máxima soportada.	Req. Funcional	Media	Habilitar la opción de reapertura del libro sacramental, si la cantidad máxima de registros permitidos es mayor a la cantidad de registros.
Secretaria	Req-011	Modificar los datos de un libro sacramental, sólo si está aperturado.	Req. Funcional	Media	Inhabilitar el control de actualización de datos del libro sacramental si está cerrado.
Secretaria	Req-012	Impedir el registro de sacramentos si no hay ningún libro de dicho sacramento aperturado.	Req. Funcional	Alta	Impedir el inicio de la interfaz del sacramento si no hay ningún libro sacramental de dicho sacramento abierto. Mostrar mensaje al usuario.
Secretaria	Req-013	Actualizar los datos específicos de la Confirmación: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Nombres de la Madre, Lugar de nacimiento, Fecha de nacimiento, Lugar de Bautismo,	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de confirmaciones, durante las pruebas de Control de Calidad

		<p>Fecha de Bautismo, Nº Libro – Fojas – Número del Bautismo, Fecha de la Confirmación, Nombre del Padrino o Madrina, Ministro o Celebrante, Anotaciones.</p> <p>Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.</p>			
Secretaria	Req-014	<p>Actualizar los datos específicos del Matrimonio: Nombre de la Parroquia, Nº Libro – Fojas – Número, Nombres del Esposo, Nacido en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Nombres de la Esposa, Nacida en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Fecha de Bautismo, Fecha del Matrimonio, Nombre del Padrino, Nombre de la Madrina, Nombre de los Testigos (2), Ministro o Celebrante, Anotaciones.</p> <p>Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.</p>	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de matrimonios, durante las pruebas de Control de Calidad
Secretaria	Req-015	<p>Posibilidad de ingresar los datos del bautismo del feligrés en la confirmación (esposo o esposa en el matrimonio) manualmente, en caso que el feligrés no tenga registros de bautismo en la parroquia.</p>	Req. Funcional	Alta	Habilitación automática de los campos correspondientes en caso de no encontrar el registro.
Secretaria	Req-016	<p>No permitir la modificación de los registros de sacramentos para los campos: feligrés y libro, folio y número.</p>	Req. Funcional	Alta	Inhabilitar los campos: feligrés, libro, folio y número, durante el

					proceso de actualización de datos de los sacramentos.
Secretaria	Req-017	Ordenar los registros del feligrés y personal alfabéticamente por apellidos.	Req. No Funcional	Media	Ordenación de manera correcta por el apellido del feligrés al momento de realizar la búsqueda.
Secretaria	Req-018	Buscar los sacramentos por el Apellido Paterno del feligrés.	Req. No Funcional	Media	Contemplar el apellido paterno como filtro de búsqueda en los sacramentos
Párroco	Req-019	Actualizar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz del personal, durante las pruebas de Control de Calidad
Párroco	Req-020	Actualizar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de las comunidades, durante las pruebas de Control de Calidad
Párroco	Req-021	Actualizar los datos de tipos de personal y de comunidades. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de tipos de personal y comunidades, durante

					las pruebas de Control de Calidad
Párroco	Req-022	Actualizar los datos de los lugares de reunión de las comunidades Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de lugares de reunión, durante las pruebas de Control de Calidad
Párroco	Req-023	Posibilidad de añadir un personal a distintas comunidades y eliminar una asignación de un personal.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de asignaciones, durante las pruebas de Control de Calidad
Secretaria	Req-024	Buscar los registros de forma rápida y precisa.	Req. No Funcional	Media	Función correcta de los filtros de búsqueda
Párroco	Req-025	Realizar copias de seguridad de la base de datos del sistema.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de backup's, durante las pruebas de Control de Calidad
Párroco	Req-026	Restablecer las copias de seguridad generadas del sistema.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de restauración, durante las pruebas de Control de Calidad

Secretaria	Req-027	Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros soportados.	Req. No Funcional	Media	Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros permitidos. Mostrar un mensaje al usuario.
Párroco	Req-028	Imprimir los datos de los sacramentos con el formato de actas de la parroquia	Req. No Funcional	Alta	Cero errores al momento de generar el pdf de las actas de sacramentos
Párroco	Req-029	Imprimir la lista de personal asignado a una comunidad	Req. No Funcional	Alta	Cero errores al momento de generar el pdf del personal asignado

Tabla 18 Requerimientos del Producto

### 3.2.2. Diseño

Diagrama de Componentes

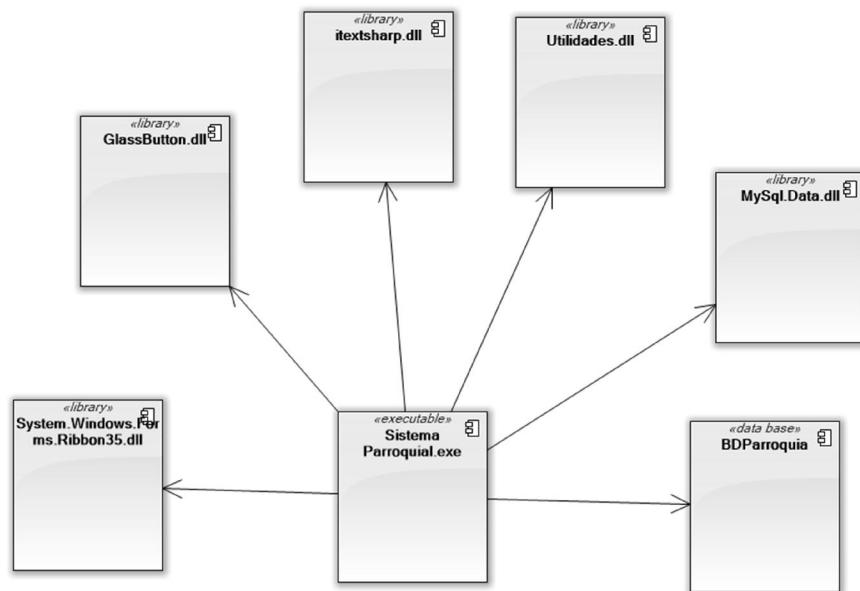


Ilustración 5 Diagrama de Componentes

Diagrama de Despliegue

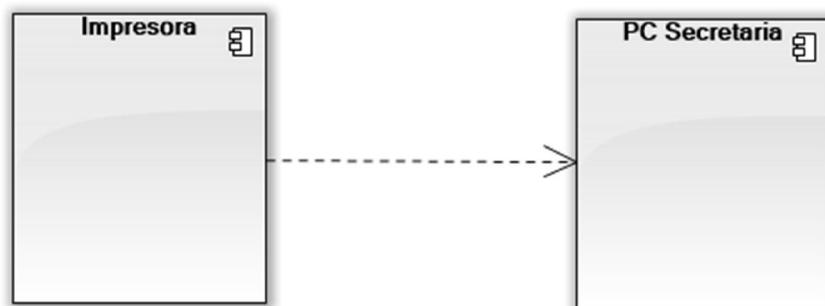


Ilustración 6 Diagrama de Despliegue

## Diagrama de Paquetes

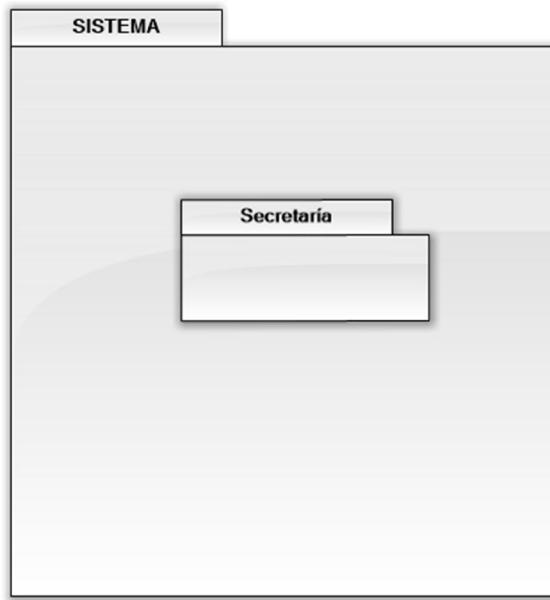


Ilustración 7 Diagrama de Paquetes

## Diagrama de Secuencia de Asignación

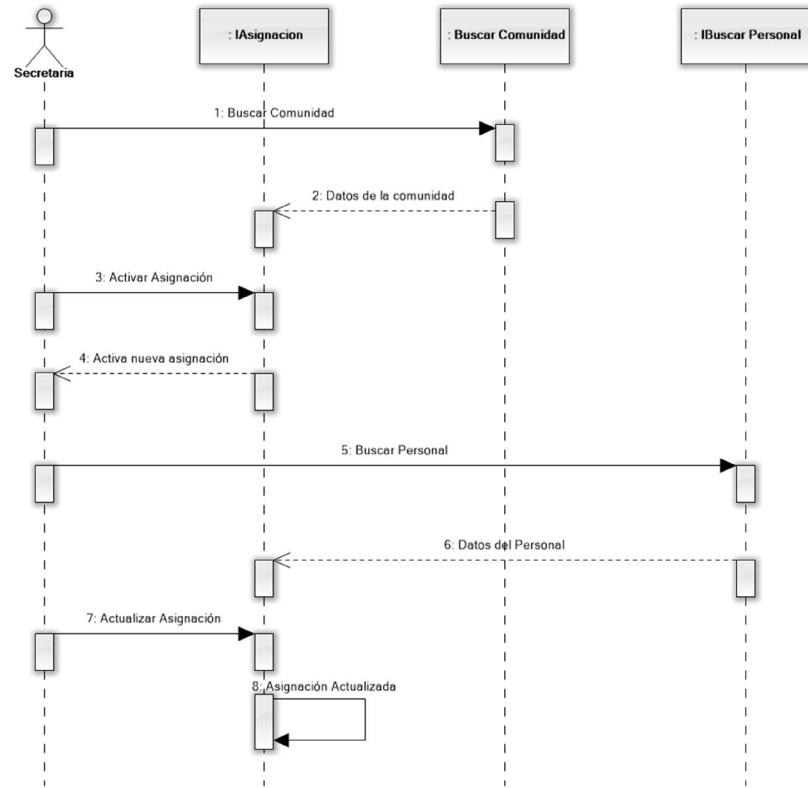
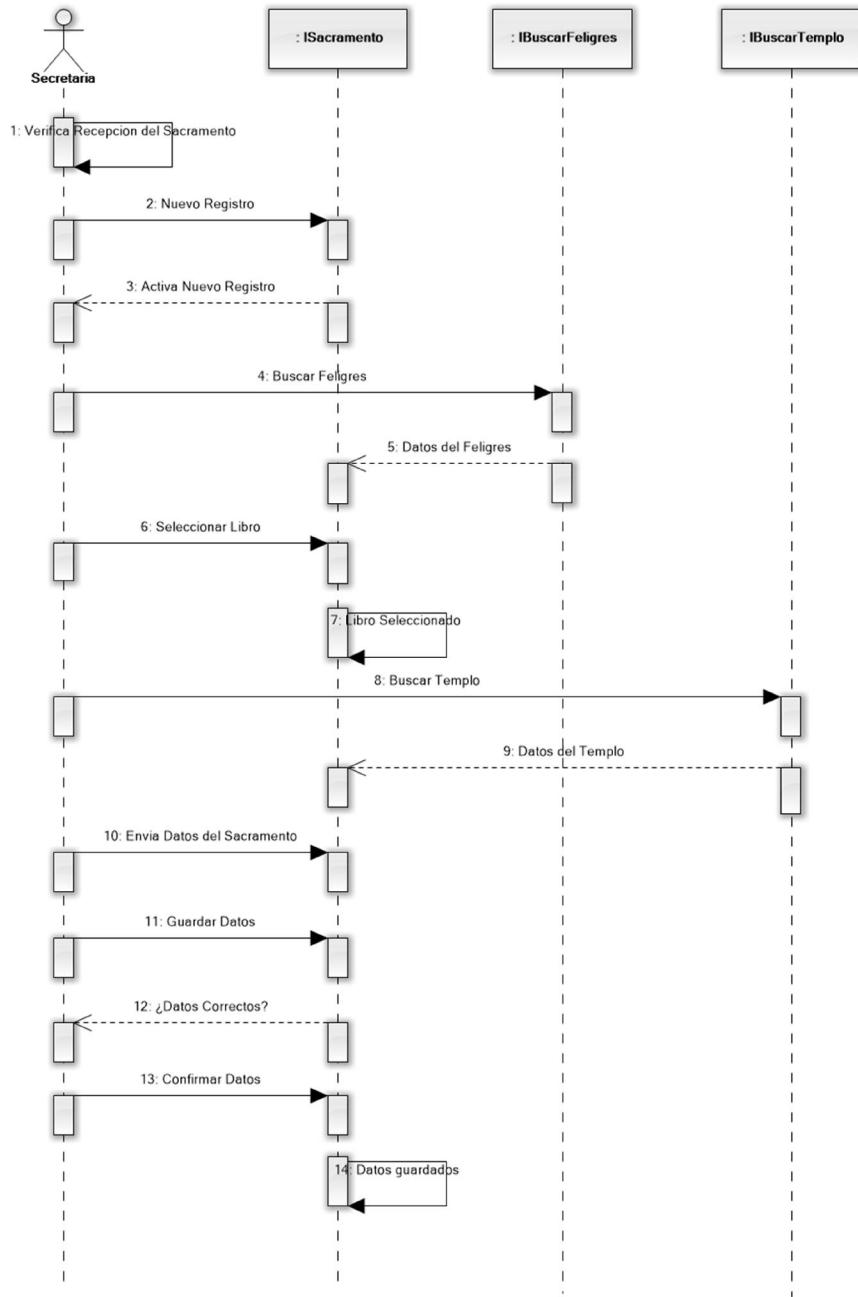


Ilustración 8 Diagrama de Secuencia de Asignación

## Diagrama de Secuencia de Sacramentos



## Ilustración 9 Diagrama de Secuencia de Sacramentos

### **3.3. Soporte del proyecto**

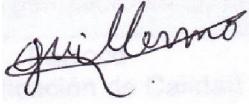
#### **3.3.1. Planificación de la Calidad**

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	JC	LL	VD	20/06/2013	Original
2.0	JC	LL	VD	30/09/2013	Actualización

### **Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal**

#### **Elaboración del Plan de Gestión de Calidad**

PLAN DE GESTIÓN DE CALIDAD	Fecha de Aprobación: 04/10/2013
<b>Descripción del Sistema de Calidad para el Proyecto</b>	
Comprende la organización para la calidad, procesos, procedimientos y documentación orientados para el logro de los objetivos del proyecto de desarrollo e implantación.	
<b>Estructura organizacional de la Calidad</b>	
<ul style="list-style-type: none"><li>- Dirección : Párroco</li><li>- Área Planificación de Calidad : Gerente de proyecto</li><li>- Área de Auditoría:<ul style="list-style-type: none"><li>o Asesor</li></ul></li></ul>	
<b>Roles y Responsabilidades</b>	
<ul style="list-style-type: none"><li>- Gerente de proyectos (D)</li><li>- Asesor (P)</li><li>- Cliente (P)</li></ul> <p>D: Dirige    P: Participa</p>	<ul style="list-style-type: none"><li>- Planificación de la calidad;</li><li>- Aseguramiento de la Calidad</li><li>- Aseguramiento de la calidad</li><li>- Aseguramiento de la calidad</li></ul>
<b>Procedimientos y Normas</b>	
<ul style="list-style-type: none"><li>- Elaboración y aprobación del Acta de Constitución</li><li>- Elaboración, revisión y aprobación del Alcance del proyecto</li><li>- Manual del PMBOK 4<sup>ta</sup> Edición</li></ul>	

<b>Procesos</b>		
<ul style="list-style-type: none"> <li>- Gestión de Proyectos</li> <li>- Inicio</li> <li>- Elaboración</li> <li>- Construcción</li> <li>- Transición</li> </ul>		
<b>Recursos</b>		
<ul style="list-style-type: none"> <li>- Asesor (1)</li> <li>- Equipo de Proyecto</li> <li>- Útiles de oficina</li> <li>- PC desktop (1) <ul style="list-style-type: none"> <li>- Impresora (1)</li> </ul> </li> </ul>		
<b>Actividades de Calidad</b>		
<b>Tipo</b>	<b>Fecha</b>	<b>Descripción</b>
Planificación	20/06/2013	Elaborar Plan de Calidad
Aseguramiento de la Calidad	21/06/2013	Revisión de los Requerimientos.
Aseguramiento de la Calidad	03/07/2013	Revisión de la Estructura Base de Datos.
	Quincena y Fin de Mes	Revisión de Código Fuente
Control de Calidad	11/11/2013	Pruebas con el Usuario Líder y Cliente
Elaborado por Juan Castillo		Aprobado por Vicente Diez
		

### 3.3.2. Identificación de estándares y métricas

#### METRICA DE CALIDAD

NOMBRE DEL PROYECTO	SIGLAS DEL PROYECTO
Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal	ISIGCSAP
<b>Métrica de:</b>	
<b>Producto</b> <input checked="" type="checkbox"/> <b>Proyecto</b> <input type="checkbox"/>	
<b>FACTOR DE CALIDAD RELEVANTE</b>	
Aprobación de los módulos del sistema	
<b>DEFINICIÓN DEL FACTOR DE CALIDAD</b>	
La aprobación de los módulos se define como la satisfacción que el cliente tiene con respecto al desarrollo del sistema. Cada módulo entregado para su prueba debe cumplir con los requisitos establecidos.	
<b>PROPÓSITO DE LA MÉTRICA</b>	
La métrica se desarrolla para identificar si las funciones del sistema cumplen con los requerimientos que el cliente estableció y por ende si este sistema ayudará a la problemática presente en la parroquia.	
<b>DEFINICIÓN OPERACIONAL</b>	
Luego de cada entrega de avance del sistema, se probarán sus funciones, así como la relación de los campos con los datos reales que se manejan. El resultado de estas pruebas se verá reflejado en las Actas de Pruebas correspondientes.	
<b>MÉTODO DE MEDICIÓN</b>	
<ol style="list-style-type: none"> <li>1. Se tomará como referencia a los casos de prueba para comprobar que efectivamente el módulo cumple con los requerimientos establecidos.</li> <li>2. La evaluación de las pruebas se dará de la siguiente manera:             <ol style="list-style-type: none"> <li>a. Todas las funciones son aprobadas =&gt; Módulo OK</li> <li>b. Al menos una función no aprobada =&gt; Módulo no cumple</li> </ol> </li> <li>3. En caso de encontrarse al menos una función desaprobatoria, se tomaran notas de la causa de los posibles fallos, empezando inmediatamente la corrección de esos errores, solventándolos a la brevedad posible.</li> </ol>	
<b>RESULTADO DESEADO</b>	
El resultado de la métrica es obtener la aprobación total de cada módulo entregado al cliente para sus pruebas.	
<b>ENLACE CON OBJETIVOS ORGANIZACIONALES</b>	
El cumplimiento de esta importante métrica es conocer de primera fuente el grado de satisfacción del cliente con respecto al desarrollo del sistema. La aprobación de las funciones de los módulos muestra que el cliente está satisfecho con el avance realizado, y que el sistema se encuentra en el camino correcto.	
<b>RESPONSABLE DEL FACTOR DE CALIDAD</b>	
La persona responsable de vigilar este factor de calidad, los resultados de la métrica y promover las mejoras en el proceso del desarrollo, es el Jefe de Proyecto.	

Tabla 19 Métrica de Calidad del Producto

## METRICA DE CALIDAD

NOMBRE DEL PROYECTO	SIGLAS DEL PROYECTO
Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal	ISIGCSAP

<b>Métrica de:</b>
<b>Producto</b> <b>Proyecto</b> <input checked="" type="checkbox"/>
<b>FACTOR DE CALIDAD RELEVANTE</b>
Medición del cumplimiento de hitos a lo largo de la vida del proyecto
<b>DEFINICIÓN DEL FACTOR DE CALIDAD</b>
El factor relevante se define identificando los hitos y sus fechas de cumplimiento a lo largo de la vida del proyecto.
<b>PROPÓSITO DE LA MÉTRICA</b>
La métrica se desarrolla para identificar los cumplimientos de los hitos e identificar posibles retrasos que perjudiquen el avance del cumplimiento del producto.
<b>DEFINICIÓN OPERACIONAL</b>
El Jefe de Proyecto actualizará las fechas hito en el cronograma, de manera quincenal y calculará el avance de los hitos, en las reuniones, obteniendo de esta forma el avance del cumplimiento de los hitos.
<b>MÉTODO DE MEDICIÓN</b>
<ol style="list-style-type: none"> <li>1. Se actualizarán los avances de acuerdo a quincenas y fines de mes.</li> <li>2. Los datos se colocarán en un documento Word y se revisará conjuntamente con el sponsor.</li> <li>3. Se tomarán acciones correctivas en caso sean necesarias, para no atrasar más el desarrollo del software.</li> </ol>
<b>RESULTADO DESEADO</b>
Hitos a considerar y sus mediciones: Hitos <= 5 días después de la fecha de cumplimiento es el 90% Hitos > 5 días después de la fecha de cumplimiento es 0%.
<b>RESPONSABLE DEL FACTOR DE CALIDAD</b>
La persona responsable de vigilar este factor de calidad, los resultados de la métrica y promover las mejoras en el proceso del desarrollo, es el Jefe de Proyecto.

[Tabla 20 Métrica de Calidad del Proyecto](#)

### 3.3.3. Diseño de formatos de aseguramiento de calidad

#### Formato de Pruebas Unitarias

<b>Modulo</b>	
<b>Desarrollador:</b>	
<b>Fecha</b>	

#	Descripción de la prueba	Número de errores detectados	Estado
---	--------------------------	------------------------------	--------

**Nota:** Si no se detectan errores, colocar '0'

<b>Interfaz</b>			
1	El diseño de la pantalla / página, <u>creada o modificada</u> sigue el estándar definido en el proyecto		
2	Al hacer resize, maximize, etc., no se altera la apariencia de la pantalla.		
3	Los campos en pantalla se habilitan y deshabilitan correctamente, según la transacción o estado de la pantalla.		
4	La longitud de los campos en pantalla corresponde a la de los campos en base de datos.		
5	Los tipos de dato de los campos en pantalla corresponden a los definidos en base de datos.		
6	Los datos se ingresan, muestran y almacenan en mayúsculas o minúsculas, según lo definido.		
7	La validación de campos mandatorios funciona correctamente para todos los casos.		
8	Los campos numéricos soportan decimales, negativos, o ceros según lo definido.		
9	Los campos de fecha validan correctamente el formato definido y el rango soportado por el lenguaje de programación y por el motor de base de datos.		
10	El desplazamiento por los componentes de pantalla con la tecla TAB es ordenado.		
11	Al cancelar o grabar una transacción, los campos en pantalla se inicializan correctamente.		
12	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		
<b>Reportes</b>			
13	La longitud de los campos de salida del reporte soporta el tamaño de todos los datos definidos.		

14	El reporte se puede generar correctamente en todos los formatos definidos (Excel, texto, pdf, etc.).		
15	El formato de reporte mantiene el estándar definido por el proyecto.		
16	El reporte se puede generar correctamente con la amplitud máxima de los filtros(es decir procesando la mayor cantidad de información que permitan los filtros mandatorios.)		
17	El reporte se puede generar correctamente ingresando un valor en todos los filtros incluso los no mandatorios.		
<b>Base de Datos</b>			
18	Los objetos nuevos (SP, Tablas, Funciones, etc.) siguen los estándares de nomenclatura definidos para el proyecto.		
19	Las consultas nuevas o modificadas han sido probadas con un volumen de información similar al del ambiente de producción. En caso de ser así, ¿se comportan según lo esperado para el proyecto (en tiempo y recursos)?		
20	Se controla o tolera si el resultado de la consulta a base de datos es vacío(aun cuando la tabla debería contener información en condiciones normales)		
<b>Otros</b>			
21	El código compila sin warnings asociados al módulo u opción probada.		
22	Los flags utilizados corresponden a 0/1 o Y/N de forma consistente.		
23	En los algoritmos u operaciones matemáticas se ha controlado que las variables contengan ceros o negativos.		
24	Luego de ser utilizados, se limpian / destruyen las variables, objetos creados.		
25	Se siguen los estándares de desarrollo definidos.		
26	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		

<b>Otros definidos por el proyecto - (Insertar filas adicionales si es necesario)</b>			
27			
28			

#### **Prueba Funcional**

*En la siguiente sección, mencionar las pruebas relacionadas con la funcionalidad programada. Indicar en la columna "Caso de Uso / Especificación Funcional".*

#	Caso de Uso / Especificación Funcional	Especificación del documento Nombre y Versión	Número de errores detectados	Estado
29				
30				
31				

Total de Errores Detectados	0
-----------------------------	---

Tabla 21 Formato de Pruebas Unitarias

## **CAPÍTULO IV: EJECUCIÓN Y SEGUIMIENTO Y CONTROL DEL PROYECTO**

## 4.1. Gestión del proyecto

### 4.1.1. Ejecución

#### A. Acta de aceptación de entregables a aprobar

##### Datos de Lugar y Fecha

Fecha / Hora:	Miércoles 03 de Julio del 2013 / 4:30 p.m.
Lugar	Secretaría de la Parroquia San Clemente - Pisco
Referencia	ACTA DE ACEPTACIÓN DE MÓDULOS A PROBAR

##### Participantes

Nombre	Área (Usuaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

##### Objetivo del Documento

El presente documento tiene como objetivo brindar conocimiento de las entregas de los módulos del sistema que tendrán que probarse para confirmar la aceptación del producto. Asimismo se indican las observaciones realizadas por el usuario.

##### Observaciones

A continuación se detalla cada uno de los módulos del sistema a probar.

Ítem:	Nombre	Descripción	Estado	Observación
01	Módulo de Feligrés	<p>Este módulo contiene las interfaces necesarias para actualizar los datos de los feligreses.</p> <p>Comprende las interfaces:</p> <ul style="list-style-type: none"><li>• Países</li><li>• Ciudades</li><li>• Provincias</li><li>• Feligrés</li></ul>	Por desarrollar	NINGUNA
02	Módulo de Sacramentos	<p>Este módulo contiene las interfaces necesarias para actualizar los datos de los sacramentos.</p> <p>Comprende las</p>	Por desarrollar	NINGUNA

		<p>interfaces:</p> <ul style="list-style-type: none"> <li>• Bautismo</li> <li>• Confirmación</li> <li>• Matrimonio</li> <li>• Libro Sacramental</li> <li>• Templos</li> </ul>		
03	Módulo de Reportes de Sacramentos	<p>Este módulo contiene la interfaz que genera un pdf para imprimir más adelante en la plantilla de actas de la parroquia.</p> <p>Comprende la interfaz:</p> <ul style="list-style-type: none"> <li>• Actas</li> </ul>	Por desarrollar	NINGUNA
04	Módulo de Personal	<p>Este módulo contiene las interfaces necesarias para actualizar los datos del personal de la parroquia.</p> <p>Comprende las interfaces:</p> <ul style="list-style-type: none"> <li>• Personal</li> <li>• Tipos</li> </ul>	Por desarrollar	NINGUNA
05	Módulo de Comunidades	<p>Este módulo contiene las interfaces necesarias para actualizar los datos de las comunidades y sus asignaciones.</p> <p>Comprende las interfaces:</p> <ul style="list-style-type: none"> <li>• Comunidades</li> <li>• Lugares de Reunión</li> <li>• Asignación</li> </ul>	Por desarrollar	NINGUNA
06	Módulo de Reportes de Asignaciones	<p>Este módulo contiene la interfaz que genera un pdf a modo de reporte en el cuál se visualiza el personal asignado a una determinada comunidad.</p> <p>Comprende la interfaz:</p> <ul style="list-style-type: none"> <li>• Personal Asignado</li> </ul>	Por desarrollar	NINGUNA

07	Módulo de Backup's	<p>Este módulo contiene las interfaces necesarias para realizar copias de seguridad de los datos del sistema y su posterior restauración. Comprende las interfaces:</p> <ul style="list-style-type: none"> <li>• Salvar Datos</li> <li>• Restaurar Datos</li> </ul>	Por desarrollar	NINGUNA
----	--------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------	---------

### Conclusiones

Se definen como totalmente aprobados los ítems del documento por los usuarios responsables

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

## B. Acta de reunión de equipo

### ACTA DE REUNIÓN

#### Datos de Reunión

Fecha y Hora :	Miércoles 17 de Julio del 2013 / 04:30 p.m.
Lugar :	Secretaría de la Parroquia San Clemente - Pisco
Organizador :	Juan Castillo

#### Participantes

Nombre	Área	Nemónico
Juan Castillo	Gerencia de Proyecto	JC
Vicente Diez	Sponsor	VD
Luisa Cajo	Líder de Usuario	LC

#### Temas Tratados

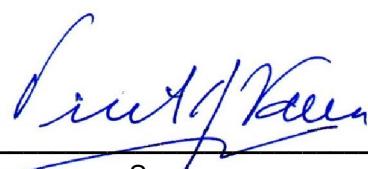
Nombre del Tema o Requerimiento :	
Entrega del primer avance del sistema	
Tema	Responsable
<b>“Entrega del primer avance del sistema”</b>	
Instalación del motor de base de datos	Juan Castillo
Instalación del módulo de feligrés	Juan Castillo
Explicación de la interfaz y uso tentativo de la misma	Juan Castillo
<b>“Acuerdos”</b>	
Firma de los documentos correspondientes	Vicente Diez
Realizar las pruebas respectivas	Luisa Cajo
Continuar con el desarrollo del proyecto respetando las fechas según cronograma	Juan Castillo

Concluida la presente reunión a las 17:00 horas, se levanta la presente acta, firmándose en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

Por la Parroquia:



Líder usuario



Sponsor

Por el Equipo de Proyecto:



Gerente de Proyecto

# ACTA DE REUNIÓN

## Datos de Reunión

Fecha y Hora :	Miércoles 21 de Agosto del 2013 / 04:10 p.m.
Lugar :	Secretaría de la Parroquia San Clemente - Pisco
Organizador :	Juan Castillo

## Participantes

Nombre	Área	Nemónico
Juan Castillo	Gerencia de Proyecto	JC
Vicente Diez	Sponsor	VD
Luisa Cajo	Líder de Usuario	LC

## Temas Tratados

Nombre del Tema o Requerimiento :	
Entrega del segundo avance del sistema	
Tema	Responsable
“Entrega del primer avance del sistema”	
Instalación del módulo de sacramentos	Juan Castillo
Instalación del módulo de reportes de sacramentos	Juan Castillo
“Acuerdos”	
Firma de los documentos correspondientes	Vicente Diez
Realizar las pruebas respectivas	Luisa Cajo
Continuar con el desarrollo del proyecto respetando las fechas según cronograma	Juan Castillo

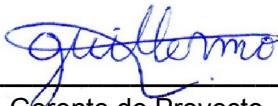
Concluida la presente reunión a las 16:30 horas, se levanta la presente acta, firmándose en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

Por la Parroquia:

  
\_\_\_\_\_  
Líder usuario

  
\_\_\_\_\_  
Sponsor

Por el Equipo de Proyecto:

  
\_\_\_\_\_  
Gerente de Proyecto

# ACTA RE REUNIÓN

## Datos de Reunión

<b>Fecha y Hora :</b>	Miércoles 18 de Septiembre del 2013 / 6:00 p.m.
<b>Lugar :</b>	Secretaría de la Parroquia San Clemente - Pisco
<b>Organizador :</b>	Vicente Diez

## Participantes

Nombre	Área	Nemónico
Juan Castillo	Gerencia de Proyecto	JC
Vicente Diez	Sponsor	VD
Luisa Cajo	Líder de Usuario	LC

## Temas Tratados

Nombre del Tema o Requerimiento :	
Cambio al cronograma de trabajo y a los requerimientos. Entrega del tercer avance del sistema	
Tema	Responsable
<b>“Cambio al cronograma de trabajo y a los requerimientos. Entrega del tercer avance del sistema”</b>	
Cambio de la fecha de término del proyecto al 21 de Noviembre del 2013	Vicente Diez
Supresión del requerimiento relacionado con la asistencia del personal a los eventos	Vicente Diez
Instalación del módulo de personal	
<b>“Acuerdos”</b>	
Revisión del impacto de los cambios en el proyecto	Juan Castillo
Elaboración de los documentos correspondientes en caso de aprobarse los cambios	Juan Castillo
Firma de los documentos correspondientes	Vicente Diez
Realizar las pruebas respectivas	Luisa Cajo
Continuar con el desarrollo del proyecto respetando las fechas según cronograma	Juan Castillo

Concluida la presente reunión a las 18:40 horas, se levanta la presente acta, firmándose en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

*Por la Parroquia:*

  
 Luisa Cajo  
 Líder usuario

  
 Vicente Diez  
 Sponsor

*Por el Equipo de Proyecto:*

  
 Juan Castillo  
 Gerente de Proyecto

## ACTA DE REUNIÓN

### Datos de Reunión

Fecha y Hora :	Lunes 30 de Septiembre del 2013 / 06:30 p.m.	
Lugar :	Secretaría de la Parroquia San Clemente - Pisco	
Organizador :	Juan Castillo	

### Participantes

Nombre	Área	Nemónico
Juan Castillo	Gerencia de Proyecto	JC
Vicente Diez	Sponsor	VD
Luisa Cajo	Líder de Usuario	LC

### Temas Tratados

Nombre del Tema o Requerimiento :		
Entrega del cuarto avance del sistema. Aceptación de las peticiones de cambios.		
Tema		Responsable
<b>“Entrega del primer avance del sistema. Aceptación de las peticiones de cambios.”</b>		
Instalación del módulo de comunidades		Juan Castillo
Aceptación de las peticiones de cambios		Juan Castillo
<b>“Acuerdos”</b>		
Firma de los documentos correspondientes		Vicente Diez
Realizar las pruebas respectivas		Luisa Cajo
Continuar con el desarrollo del proyecto respetando las fechas según cronograma		Juan Castillo

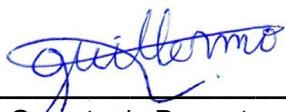
Concluida la presente reunión a las 18:45 horas, se levanta la presente acta, firmándose en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

*Por la Parroquia:*

  
\_\_\_\_\_  
Líder usuario

  
\_\_\_\_\_  
Sponsor

*Por el Equipo de Proyecto:*

  
\_\_\_\_\_  
Gerente de Proyecto

## ACTA DE REUNIÓN

### Datos de Reunión

Fecha y Hora :	Lunes 14 de Octubre del 2013 / 06:20 p.m.	
Lugar :	Secretaría de la Parroquia San Clemente - Pisco	
Organizador :	Juan Castillo	

### Participantes

Nombre	Área	Nemónico
Juan Castillo	Gerencia de Proyecto	JC
Vicente Diez	Sponsor	VD
Luisa Cajo	Líder de Usuario	LC

### Temas Tratados

Nombre del Tema o Requerimiento :		
Entrega del quinto avance del sistema		
Tema		Responsable
“Entrega del primer avance del sistema”		
Instalación del módulo de asignación		Juan Castillo
Instalación del módulo de reportes de asignación		
“Acuerdos”		
Firma de los documentos correspondientes		Vicente Diez

Realizar las pruebas respectivas	Luisa Cajo
Continuar con el desarrollo del proyecto respetando las fechas según cronograma	Juan Castillo

Concluida la presente reunión a las 18:45 horas, se levanta la presente acta, firmándose en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

*Por la Parroquia:*



\_\_\_\_\_  
Líder usuario



\_\_\_\_\_  
Sponsor

*Por el Equipo de Proyecto:*



\_\_\_\_\_  
Gerente de Proyecto

## ACTA DE REUNIÓN

### Datos de Reunión

Fecha y Hora :	Sábado 02 de Noviembre del 2013 / 12:30 p.m.
Lugar :	Secretaría de la Parroquia San Clemente - Pisco
Organizador :	Juan Castillo

### Participantes

Nombre	Área	Nemónico
Juan Castillo	Gerencia de Proyecto	JC
Vicente Diez	Sponsor	VD
Luisa Cajo	Líder de Usuario	LC

### Temas Tratados

**Nombre del Tema o Requerimiento :**

Seguridad en el acceso al sistema

Tema	Responsable
<b>"Seguridad en el acceso al sistema"</b>	
Login para acceder al sistema	Juan Castillo
Tipos de login: Administrador y Usuario	Juan Castillo
Ventana para el mantenimiento de usuarios	Juan Castillo
<b>"Acuerdos"</b>	
Queda consideración de la parroquia el implementar el acceso al sistema, el tipo de acceso y la ventana para el mantenimiento de usuarios	Vicente Diez

Concluida la presente reunión a las 13:00 horas, se levanta la presente acta, firmándose en señal de conformidad por los participantes, sin perjuicio de posteriores reuniones y/o comunicaciones sobre el particular.

*Por la Parroquia:*

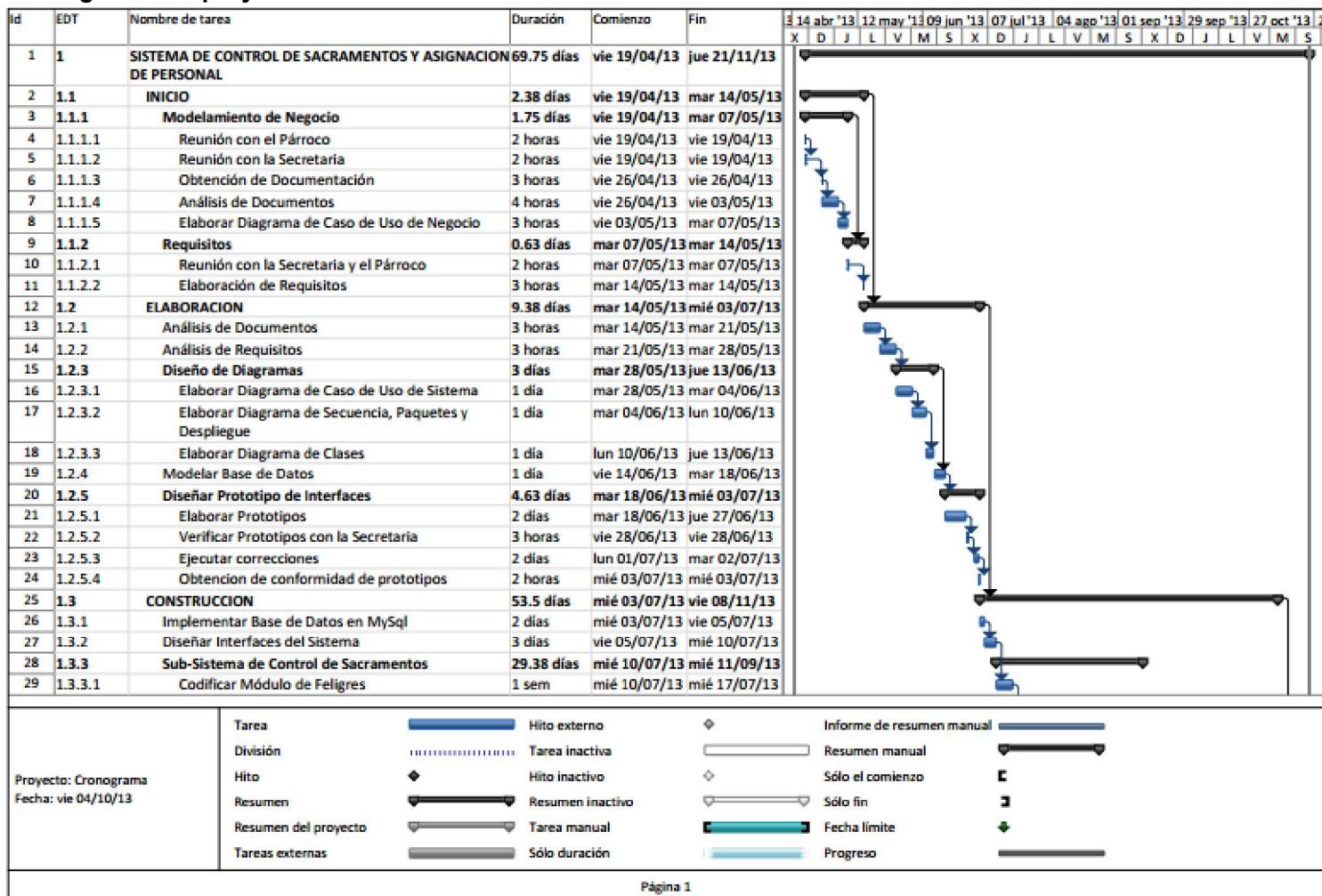
  
Líder usuario

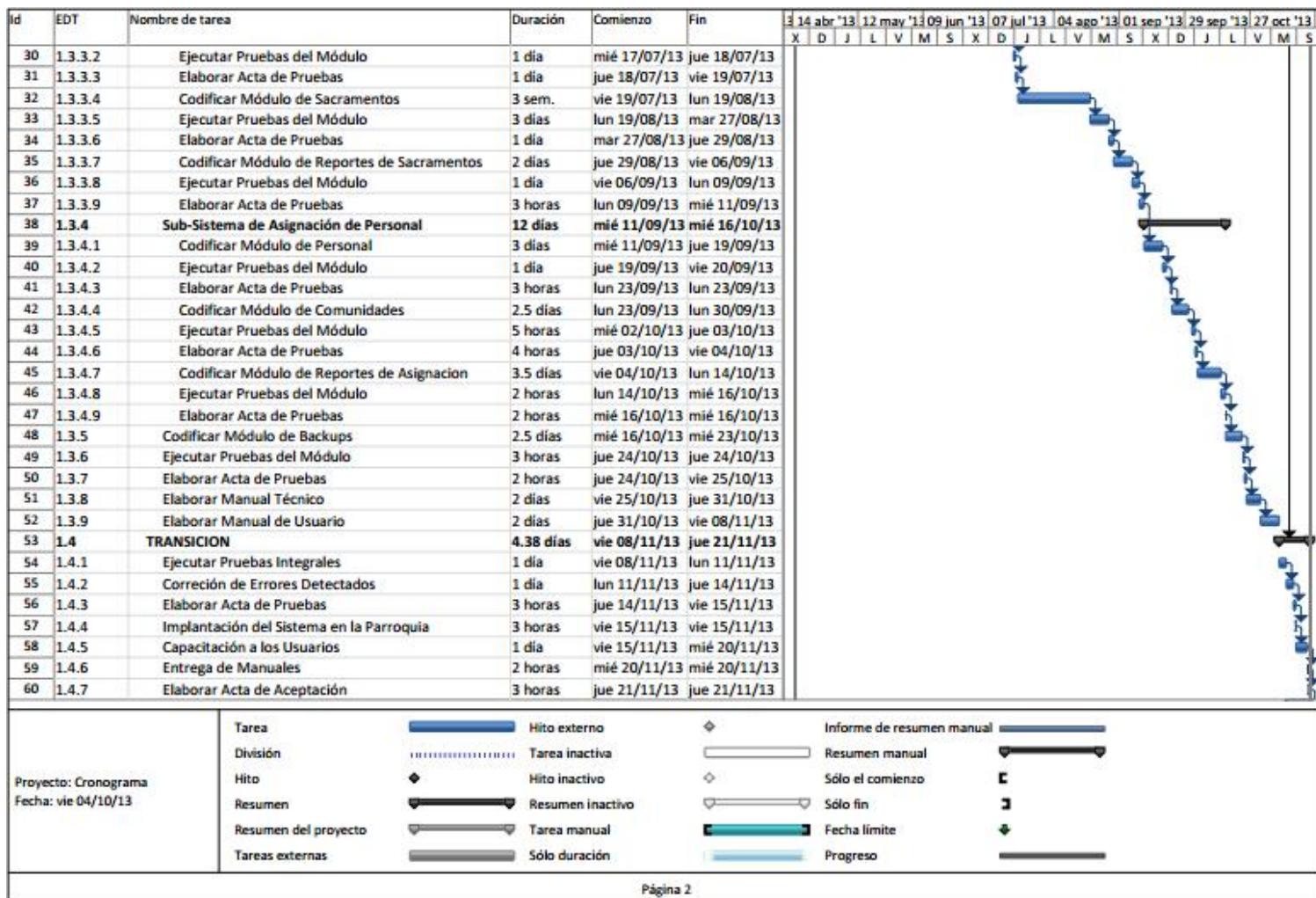
  
Sponsor

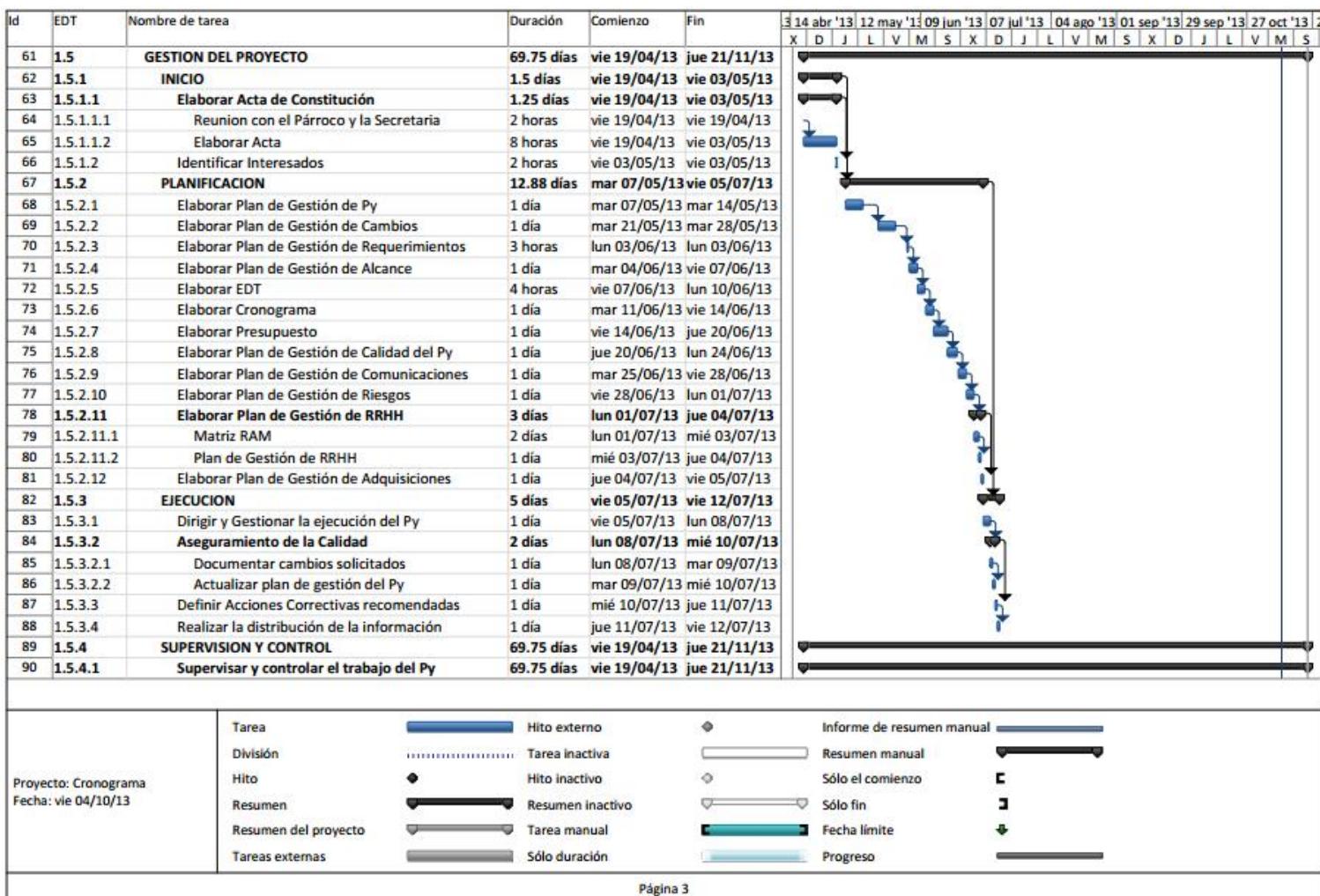
*Por el Equipo de Proyecto:*

  
Gerente de Proyecto

### C. Cronograma del proyecto









Id	EDT	Nombre de tarea	Duración	Comienzo	Fin	Cronograma														
						314 abr '13	12 may '13	09 jun '13	07 jul '13	04 ago '13	01 sep '13	29 sep '13	27 oct '13	03 nov '13	07 dic '13	11 ene '14	15 feb '14	21 mar '14		
X	D	J	L	V	M	S	X	D	J	L	V	M	S	X	D	J	L	V	M	S
115	1.5.4.7.1	Documentar riesgos surgidos	69.75 días	vie 19/04/13	jue 21/11/13															
116	1.5.4.7.2	Documentar acciones realizadas	69.75 días	vie 19/04/13	jue 21/11/13															
117	1.5.4.8	<b>Control de Calidad</b>	69.75 días	vie 19/04/13	jue 21/11/13															
118	1.5.4.8.1	Documentar la reparación de defectos recomendada	69.75 días	vie 19/04/13	jue 21/11/13															
119	1.5.4.8.2	Documentar los productos entregables validados	69.75 días	vie 19/04/13	jue 21/11/13															
120	1.5.4.8.3	Definir Acciones Preventivas recomendadas	69.75 días	vie 19/04/13	jue 21/11/13															
121	1.5.5	<b>CIERRE</b>	2 días	vie 15/11/13	jue 21/11/13															
122	1.5.5.1	Cierre del Py	1 día	vie 15/11/13	lun 18/11/13															
123	1.5.5.1.1	Cierre Administrativo	1 día	vie 15/11/13	lun 18/11/13															
124	1.5.5.1.1.1	Lecciones Aprendidas	1 día	vie 15/11/13	lun 18/11/13															
125	1.5.5.1.1.1.	Documentar Lecciones Aprendidas	1 día	vie 15/11/13	lun 18/11/13															
126	1.5.5.2	Cierre del Contrato	1 día	lun 18/11/13	jue 21/11/13															
127	1.5.5.2.1	Aceptación Final del Py	1 día	lun 18/11/13	jue 21/11/13															
128	1.5.5.2.1.1	Entregar Producto Final	1 día	lun 18/11/13	jue 21/11/13															

Proyecto: Cronograma Fecha: vie 04/10/13	Tarea		Hito externo	<input type="radio"/>	Informe de resumen manual	
	División		Tarea inactiva	<input type="radio"/>	Resumen manual	
	Hito	<input checked="" type="radio"/>	Hito inactivo	<input type="radio"/>	Sólo el comienzo	
	Resumen		Resumen inactivo	<input type="radio"/>	Sólo fin	
	Resumen del proyecto		Tarea manual	<input checked="" type="radio"/>	Fecha límite	
	Tareas externas		Sólo duración	<input type="radio"/>	Progreso	

## D. Hoja de costos

<b>EDT</b>	<b>Actividad</b>	<b>Duració n</b>	<b>UM Duració n</b>	<b>Cost o Unit.</b>	<b>Costo Total</b>	<b>Línea Base</b>	<b>Variación</b>	<b>Real</b>	<b>Restantes</b>
1.1.1	Modelamiento de Negocio	1.75	día		S/. 182.00	S/. 182.00	S/. 0.00	S/. 0.00	S/. 182.00
1.1.2	Requisitos	0.63	día		S/. 38.00	S/. 38.00	S/. 0.00	S/. 0.00	S/. 38.00
1.2.1	Análisis de Documentos	3	hora		S/. 36.00	S/. 36.00	S/. 0.00	S/. 0.00	S/. 36.00
1.2.2	Análisis de Requisitos	3	hora		S/. 36.00	S/. 36.00	S/. 0.00	S/. 0.00	S/. 36.00
1.2.3	Diseño de Diagramas	3	día		S/. 192.00	S/. 192.00	S/. 0.00	S/. 0.00	S/. 192.00
1.2.4	Modelar Base de Datos	1	día		S/. 96.00	S/. 96.00	S/. 0.00	S/. 0.00	S/. 96.00
1.2.5	Diseñar Prototipo de Interfaces	4.63	día		S/. 437.00	S/. 437.00	S/. 0.00	S/. 0.00	S/. 437.00
1.3.1	Implementar Base de Datos en MySql	2	día		S/. 362.00	S/. 362.00	S/. 0.00	S/. 0.00	S/. 362.00
1.3.2	Diseñar Interfaces del Sistema	3	día		S/. 769.00	S/. 769.00	S/. 0.00	S/. 0.00	S/. 769.00
1.3.3	Sub-Sistema de Control de Sacramentos	29.38	día		S/. 3,036.00	S/. 3,036.00	S/. 0.00	S/. 0.00	S/. 3,036.00
1.3.4	Sub-Sistema de Asignación de Personal	12	día		S/. 1,641.00	S/. 1,641.00	S/. 0.00	S/. 0.00	S/. 1,641.00
1.3.5	Codificar Módulo de Backups	2.5	día		S/. 420.00	S/. 420.00	S/. 0.00	S/. 0.00	S/. 420.00
1.3.6	Ejecutar Pruebas del Módulo	3	hora		S/. 23.00	S/. 23.00	S/. 0.00	S/. 0.00	S/. 23.00
1.3.7	Elaborar Acta de Pruebas	2	hora		S/. 42.00	S/. 42.00	S/. 0.00	S/. 0.00	S/. 42.00
1.3.8	Elaborar Manual Técnico	2	día		S/. 206.00	S/. 206.00	S/. 0.00	S/. 0.00	S/. 206.00
1.3.9	Elaborar Manual de Usuario	2	día		S/. 206.00	S/. 206.00	S/. 0.00	S/. 0.00	S/. 206.00
1.4.1	Ejecutar Pruebas Integrales	1	día		S/. 53.00	S/. 53.00	S/. 0.00	S/. 0.00	S/. 53.00
1.4.2	Corrección de Errores Detectados	1	día		S/. 288.00	S/. 288.00	S/. 0.00	S/. 0.00	S/. 288.00
1.4.3	Elaborar Acta de Pruebas	3	hora		S/. 48.00	S/. 48.00	S/. 0.00	S/. 0.00	S/. 48.00

1.4.4	Implantación del Sistema en la Parroquia	3 hora		S/. 35.00	S/. 35.00	S/. 0.00	S/. 0.00	S/. 35.00
1.4.5	Capacitación a los Usuarios	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
1.4.6	Entrega de Manuales	2 hora		S/. 32.00	S/. 32.00	S/. 0.00	S/. 0.00	S/. 32.00
1.4.7	Elaborar Acta de Aceptación	3 hora		S/. 48.00	S/. 48.00	S/. 0.00	S/. 0.00	S/. 48.00
1.5.1.1	Elaborar Acta de Constitución	1.25 día		S/. 112.00	S/. 112.00	S/. 0.00	S/. 0.00	S/. 112.00
1.5.1.2	Identificar Interesados	2 hora		S/. 12.00	S/. 12.00	S/. 0.00	S/. 0.00	S/. 12.00
1.5.2.1	Elaborar Plan de Gestión de Py	1 día		S/. 80.00	S/. 80.00	S/. 0.00	S/. 0.00	S/. 80.00
1.5.2.2	Elaborar Plan de Gestión de Cambios	1 día		S/. 80.00	S/. 80.00	S/. 0.00	S/. 0.00	S/. 80.00
1.5.2.3	Elaborar Plan de Gestión de Requerimientos	3 hora		S/. 48.00	S/. 48.00	S/. 0.00	S/. 0.00	S/. 48.00
1.5.2.4	Elaborar Plan de Gestión de Alcance	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
1.5.2.5	Elaborar EDT	4 hora		S/. 24.00	S/. 24.00	S/. 0.00	S/. 0.00	S/. 24.00
1.5.2.6	Elaborar Cronograma	1 día		S/. 48.00	S/. 48.00	S/. 0.00	S/. 0.00	S/. 48.00
1.5.2.7	Elaborar Presupuesto	1 día		S/. 80.00	S/. 80.00	S/. 0.00	S/. 0.00	S/. 80.00
1.5.2.8	Elaborar Plan de Gestión de Calidad del Py	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
1.5.2.9	Elaborar Plan de Gestión de Comunicaciones	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
1.5.2.10	Elaborar Plan de Gestión de Riesgos	1 día		S/. 80.00	S/. 80.00	S/. 0.00	S/. 0.00	S/. 80.00
1.5.2.11	Elaborar Plan de Gestión de RRHH	3 día		S/. 240.00	S/. 240.00	S/. 0.00	S/. 0.00	S/. 240.00
1.5.2.12	Elaborar Plan de Gestión de Adquisiciones	1 día		S/. 176.00	S/. 176.00	S/. 0.00	S/. 0.00	S/. 176.00
1.5.3.1	Dirigir y Gestionar la ejecución del Py	1 día		S/. 176.00	S/. 176.00	S/. 0.00	S/. 0.00	S/. 176.00
1.5.3.2	Aseguramiento de la Calidad	2 día		S/. 208.00	S/. 208.00	S/. 0.00	S/. 0.00	S/. 208.00

1.5.3.3	Definir Acciones Correctivas recomendadas	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
1.5.3.4	Realizar la distribución de la información	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
1.5.4.1	Supervisar y controlar el trabajo del Py	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.2	Control Integrado de Cambios	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.3	Verificación del Alcance	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.4	Control del Alcance	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.5	Control del Cronograma	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.6	Control de Costos	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.7	Control y Seguimiento de Riesgos	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.4.8	Control de Calidad	69.75 día		S/. 152.00	S/. 152.00	S/. 0.00	S/. 0.00	S/. 152.00
1.5.5.1.1.1	Lecciones Aprendidas	1 día		S/. 176.00	S/. 176.00	S/. 0.00	S/. 0.00	S/. 176.00
1.5.5.2.1	Aceptación Final del Py	1 día		S/. 128.00	S/. 128.00	S/. 0.00	S/. 0.00	S/. 128.00
<b>Total de Costo del Proyecto</b>				S/. 11,878.00	S/. 11,878.00	S/. 0.00	S/. 0.00	S/. 11,878.00
<b>Total de Costo de Reserva de Contingencia</b>								S/. 1,187.80
<b>Total de Costo de Reserva de Gestión</b>								S/. 13,065.80

Tabla 22 Hoja de Costos

#### 4.1.2. Seguimiento y control

##### A. Matriz de trazabilidad de requerimientos

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1	JGCH	LLP	VDV	14/05/2013	versión original

MATRIZ DE TRAZABILIDAD DE REQUERIMIENTOS

NOMBRE DEL PROYECTO		SIGLAS DEL PROYECTO	
Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal		ISIGCSAP	

ESTADO ACTUAL	
Estado	Abreviatura
Activo	AC
Cancelado	CA
Diferido	DI
Adicionado	AD
Aprobado	AP

NIVEL DE ESTABILIDAD	
Estado	Abreviatura
Alto	A
Mediano	M
Bajo	B

GRADO DE COMPLEJIDAD	
Estado	Abreviatura
Alto	A
Mediano	M
Bajo	B

CÓDIGO	DESCRIPCIÓN	ATRIBUTOS DE REQUERIMIENTO									TRAZABILIDAD HACIA:								
		SUSTENTO DE SU INCLUSIÓN	PROPIETARIO	FUENTE	PRIORIDAD	VERSIÓN	ESTADO ACTUAL	FECHA DE CUMPLIMIENTO	NIVEL DE ESTABILIDAD	GRADO DE COMPLEJIDAD	CRITERIO DE ACEPTACIÓN	NECESIDADES, OPORTUNIDADES, METAS Y OBJETIVOS DEL NEGOCIO	OBJETIVOS DEL PROYECTO	ALCANCE DEL PROYECTO /ENTREGABLE DEL WBS	DISEÑO DEL SISTEMA	DESARROLLO DEL SISTEMA	ESTRATEGIA DE PRUEBA	ESCENARIO DE PRUEBA	REQUERIMIENTO DE ALTO NIVEL
Req-001	Actualizar los datos de los países. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Apoyo al mantenimiento de los feligreses y templos	Parroquia San Clemente		Media	1.00	AC	19/07/2013	M	M	Firma de acta de pruebas del módulo Feligrés.	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	No aplica.	Cumplir con el desarrollo del sistema.
Req-002	Actualizar los datos de las ciudades o departamentos. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Apoyo al mantenimiento de los feligreses y templos	Parroquia San Clemente		Media	1.00	AC	19/07/2013	M	M	Firma de acta de pruebas del módulo Feligrés	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	No aplica.	Cumplir con el desarrollo del sistema
Req-003	Actualizar los datos de las provincias del Perú. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Apoyo al mantenimiento de los feligreses y templos	Parroquia San Clemente		Media	1.00	AC	19/07/2013	M	M	Firma de acta de pruebas del módulo Feligrés	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	No aplica.	Cumplir con el desarrollo del sistema
Req-004	Para los países, ciudades o departamento y evitar posibles fallos al momento de registrar los usuarios.	Evitar posibles fallos al momento de registrar los usuarios.	Parroquia San Clemente		Media	1.00	AC	19/07/2013	M	M	Firma de acta de pruebas del módulo Feligrés	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Intentar eliminar algún registro por defecto.	Cumplir con el desarrollo del sistema

	provincias, sólo se podrán eliminar los nuevos registros que se ingresen en el sistema.																
Req-005	Registrar los datos de los templos; Nombre, tipo, lugar. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Parroquia San Clemente	Secretaria lo solicita	Alta	1,00	AC	28/08/2013	A	A	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica.	Registrar un templo y/o actualizar sus datos.	Cumplir con el desarrollo del sistema
Req-006	Registrar los datos del feligrés. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Parroquia San Clemente	Secretaria lo solicita	Muy Alta	1,00	AC	02/08/2013	A	A	Firma de acta de pruebas del módulo Feligrés	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica.	Registrar un feligrés y/o actualizar sus datos.	Cumplir con el desarrollo del sistema
Req-007	Posibilidad de registrar a un personal como Personal y viceversa.	Parroquia San Clemente	Secretaria lo solicita	Alta	1,00	AC	23/09/2013	A	A	Firma de acta de pruebas del módulo Personal	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica.	Probar la funcionalidad de la casilla "Existe como Personal" y "Existe como Feligrés".	Cumplir con el desarrollo del sistema
Req-008	Registrar los datos del Bautismo. Nombre de la Parroquia, Nº Libro – Folios – Número, Apellidos y Nombres, Nombre del Padre, Natural del Padre, Nombres de la Madre, Natural de la Madre, Lugar de nacimiento, Fecha de nacimiento, Fecha de Bautismo, Nombre del Padrino, Nombre de la Madrina, Ministro o Celebrante, Nota Marginal. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Parroquia San Clemente	Secretaria lo solicita	Muy Alta	1,00	AC	28/08/2013	A	A	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica.	Registrar un bautismo y/o actualizar sus datos.	Cumplir con el desarrollo del sistema
Req-009	Registrar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios, Nº de registros por folio, fecha de apertura, fecha de cierre, fecha de cierre, estado, cantidad de registros. Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reabrir libro, Buscar y Cancelar.	Parroquia San Clemente	Secretaria lo solicita	Muy Alta	1,00	AC	28/08/2013		A	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica.	Aperturar un Libro Sacramental y/o actualizar sus datos	Cumplir con el desarrollo del sistema
Req-010	Permitir la reapertura de un libro sacramental si el total de registros no es igual a la cantidad máxima soportada.	Parroquia San Clemente	Secretaria lo solicita	Alta	1,00	AC	28/08/2013	M	M	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica.	Cerrar un Libro Sacramental manualmente y reaperturarlo.	Cumplir con el desarrollo del sistema

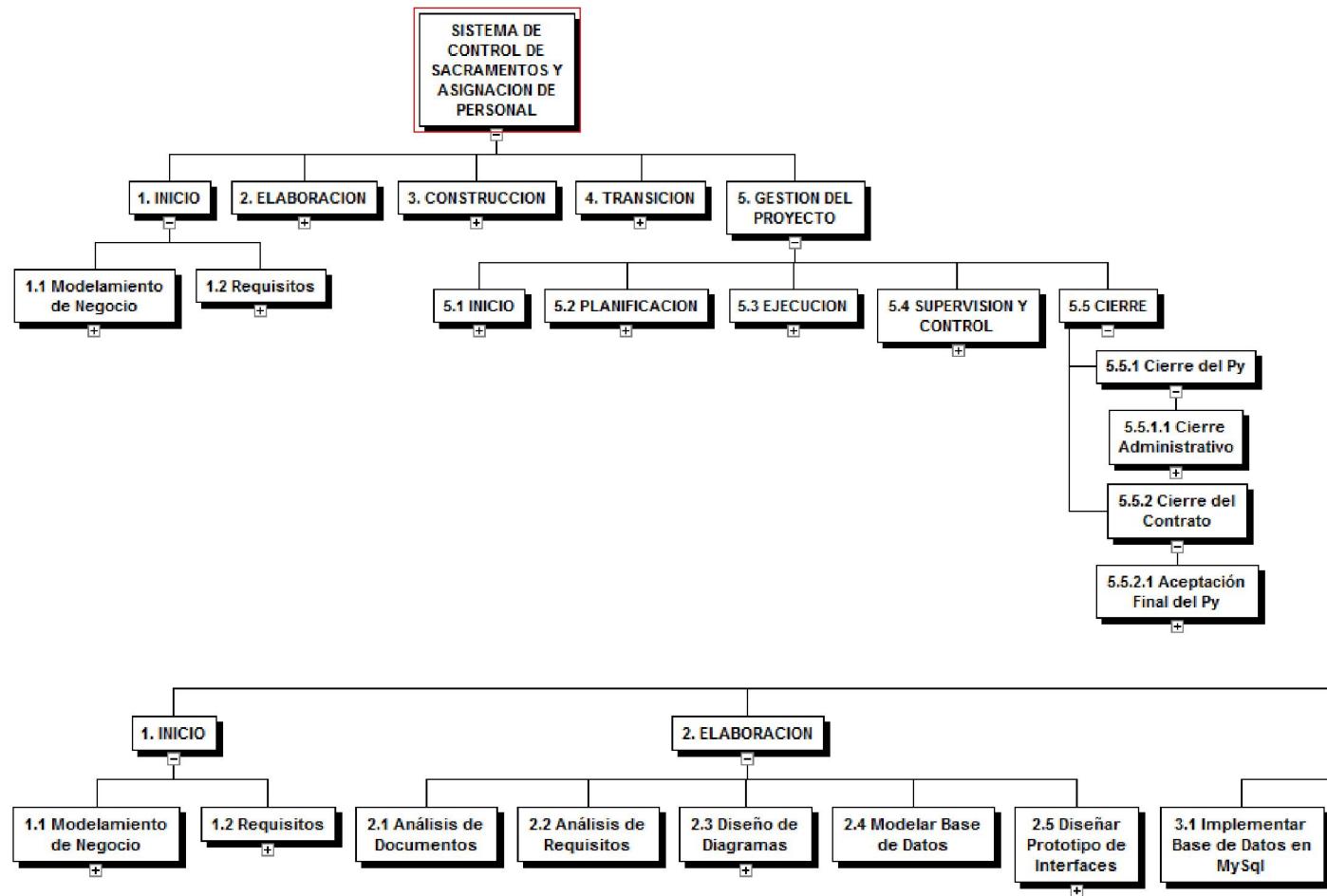
Req-011	Modificar los datos de un libro sacramental, solo si está aperturado.	Estar pendiente al momento de contabilizar los registros de los libros.	Parroquia San Clemente		Alta	1,00	AC	28/08/2013	A	M	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Cerrar un Libro Sacramental manualmente e intentar modificarlo.	Cumplir con el desarrollo del sistema
Req-012	Impedir el registro de sacramentos si no hay ningún libro de dicho sacramento aperturado.	Seguridad al momento de registrar.	Parroquia San Clemente		Alta	1,00	AC	28/08/2013	A	M	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Cerrar los libros aperturados e intentar abrir la interfaz de algún sacramento.	Cumplir con el desarrollo del sistema
Req-013	Registrar los datos específicos de la Confirmación. Nombre de la Parroquia, N° Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Nombres de la Madre, Lugar de nacimiento, Fecha de nacimiento, Lugar de Bautismo, Fecha de Bautismo, N° Libro – Fojas – Número del Bautismo, Fecha de la Confirmación, Nombre del Padrino o Madrina, Ministro o Celebrante, Anotaciones. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Secretaria lo solicita	Parroquia San Clemente		Muy Alta	1,00	AC	28/08/2013	A	A	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Registrar una confirmación y/o actualizar sus datos	Cumplir con el desarrollo del sistema
Req-014	Registrar los datos específicos del Matrimonio: Nombre de la Parroquia, N° Libro – Fojas – Número, Nombres del Esposa, Nacido en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Nombre de la Esposa, Nacida en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Fecha de Bautismo, Fecha del Matrimonio, Nombre del Padrino, Nombre de la Madrina, Nombre de los Testigos (2), Ministro o Celebrante, Anotaciones. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Secretaria lo solicita	Parroquia San Clemente		Muy Alta	1,00	AC	28/08/2013	A	A	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Registrar un matrimonio y/o actualizar sus datos	Cumplir con el desarrollo del sistema
Req-015	Possibilidad de ingresar los datos del bautismo del feligrés en la confirmación (esposo o esposa en el matrimonio) manualmente, en caso que el feligrés no tenga registros de bautismo en la parroquia.	Secretaria lo solicita	Parroquia San Clemente		Alta	1,00	AC	28/08/2013	A	M	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Intentar registrar la confirmación o matrimonio de un feligrés que no tenga bautismo registrado.	Cumplir con el desarrollo del sistema

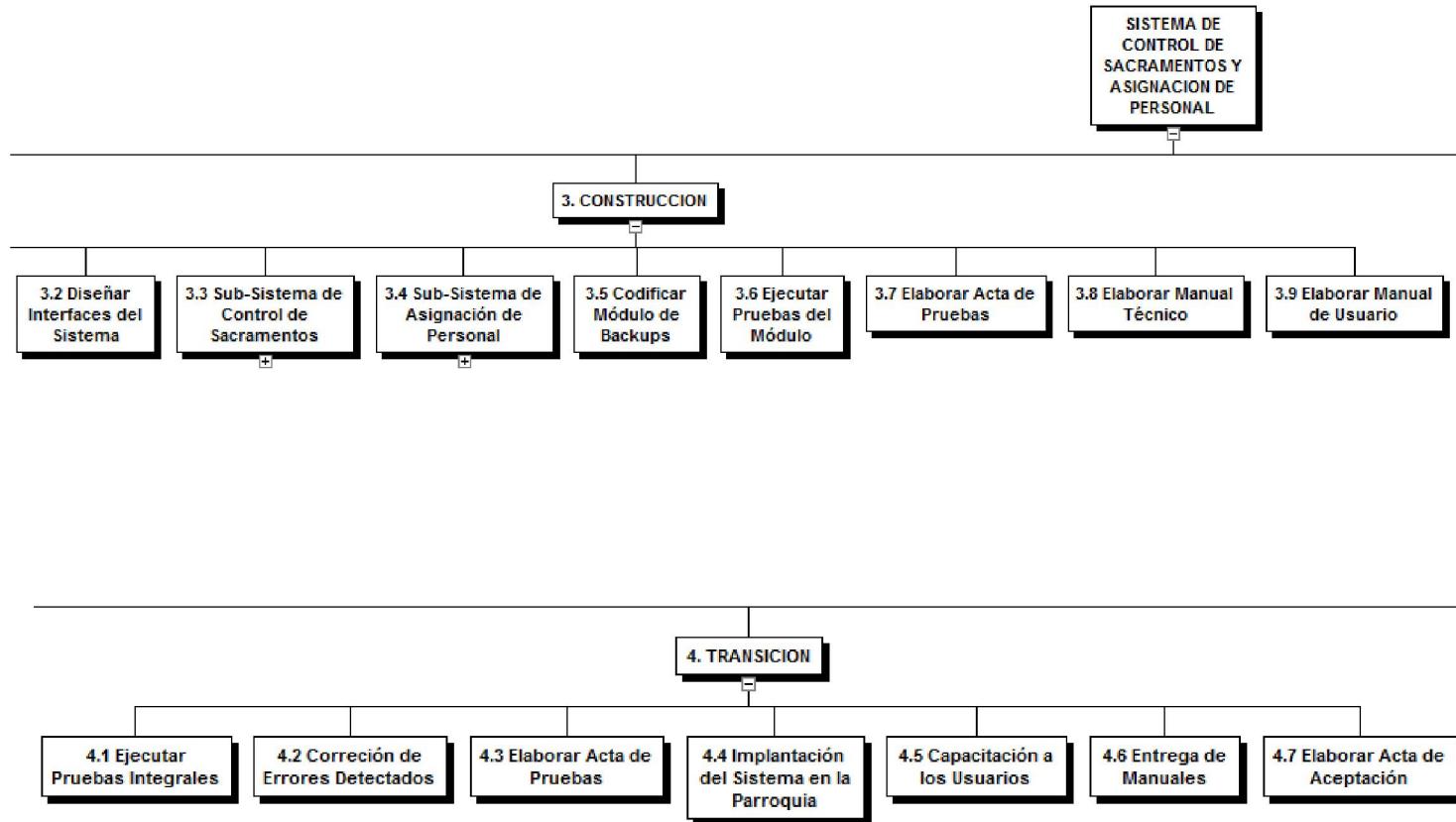
Req-016	No permitir la modificación de los registros de sacramentos para los campos: feligrés y libro, folio, número del libro.	Secretaría lo solicita	Parroquia San Clemente		Alta	1,00	AC	28/08/2013	A	M	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Intentar modificar los datos de algún sacramento	Cumplir con el desarrollo del sistema
Req-017	Ordenar los registros de feligrés y personal alfabéticamente por apellidos.	Secretaría lo solicita	Parroquia San Clemente		Media	1,00	AC	23/09/2013	M	M	Firma de acta de pruebas del módulo Personal	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Filtro de búsqueda	Cumplir con el desarrollo del sistema
Req-018	Buscar los sacramentos por el Apellido Paterno del feligrés.	Secretaría lo solicita	Parroquia San Clemente		Media	1,00	AC	28/08/2013	M	B	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Filtro de búsqueda	Cumplir con el desarrollo del sistema
Req-019	Registrar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, N° de Celular, N° de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Párroco lo solicita	Parroquia San Clemente		Alta	1,00	AC	23/09/2013	A	M	Firma de acta de pruebas del módulo Personal	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Registrar un personal y/o actualizar sus datos	Cumplir con el desarrollo del sistema
Req-020	Registrar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Párroco lo solicita	Parroquia San Clemente		Alta	1,00	AC	04/10/2013	A	M	Firma de acta de pruebas del módulo Comunidades	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Registrar una comunidad y/o actualizar sus datos	Cumplir con el desarrollo del sistema
Req-021	Actualizar los datos de tipos de personal y de comunidades. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Apoyo al mantenimiento del personal y comunidades	Parroquia San Clemente		Media	1,00	AC	23/09/2013	A	B	Firma de acta de pruebas del módulo Personal	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Registrar un tipo de personal o comunidad	Cumplir con el desarrollo del sistema
Req-022	Actualizar los datos de los lugares de reunión de las comunidades Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Apoyo al mantenimiento de las comunidades	Parroquia San Clemente		Media	1,00	AC	04/10/2013	M	B	Firma de acta de pruebas del módulo Comunidades	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Registrar un lugar de reunión de comunidad	Cumplir con el desarrollo del sistema
Req-023	Posibilidad de añadir un personal a distintas comunidades y eliminar una asignación de un personal.	Párroco lo solicita	Parroquia San Clemente		Alta	1,00	AC	04/10/2013	A	M	Firma de acta de pruebas del módulo Comunidades	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Probar interfaz de asignaciones	Cumplir con el desarrollo del sistema
Req-024	Buscar los registros de forma rápida y precisa.	Secretaría lo solicita	Parroquia San Clemente		Media	1,00	AC	19/07/2013	M	B	Firma de acta de pruebas del módulo Feligrés	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION			No aplica.	Filtro de búsqueda	Cumplir con el desarrollo del sistema

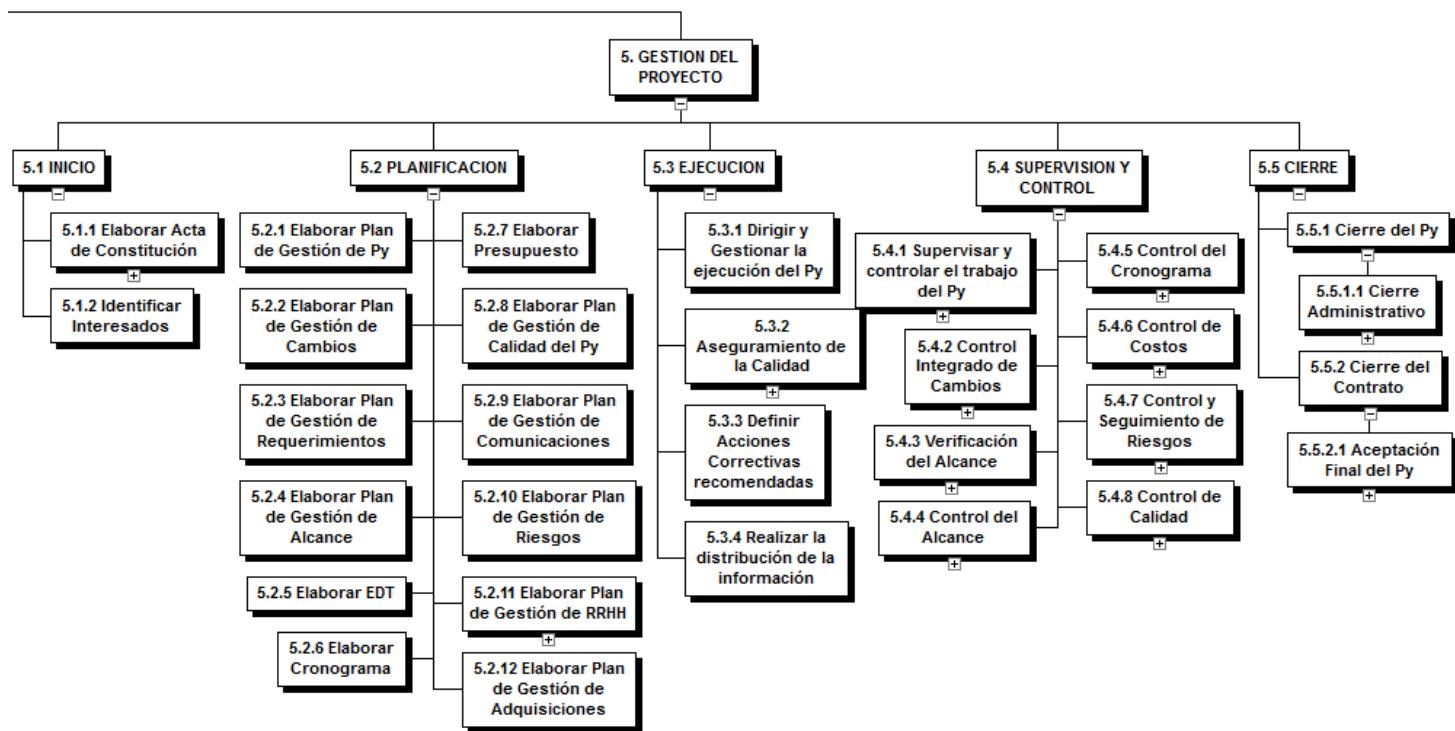
Req-025	Realizar copias de seguridad de la base de datos del sistema.	Secretaría lo solicita	Parroquia San Clemente		Muy Alta	1,00	AC	16/10/2013	A	A	Firma de acta de pruebas del módulo Backup's	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica. Probar interfaz de Salvar Datos Cumplir con el desarrollo del sistema
Req-026	Restablecer las copias de seguridad generadas del sistema.	Secretaría lo solicita	Parroquia San Clemente		Muy Alta	1,00	AC	16/10/2013	A	A	Firma de acta de pruebas del módulo Backup's	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica. Probar interfaz de Restaurar Datos Cumplir con el desarrollo del sistema
Req-027	Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros soportados.	Secretaría lo solicita	Parroquia San Clemente		Alta	1,00	AC	28/08/2013	A	M	Firma de acta de pruebas del módulo Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica. Registrar un libro sacramental de prueba de 1 registro como mínimo y registrar un sacramento Cumplir con el desarrollo del sistema
Req-028	Imprimir los datos de los sacramentos con el formato de actas de la parroquia	Secretaría lo solicita	Parroquia San Clemente		Alta	1,00	AC	28/08/2013	A	A	Firma de acta de pruebas del módulo Reportes de Sacramentos	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica. Probar la interfaz de Actas Cumplir con el desarrollo del sistema
Req-029	Imprimir la lista de personal asignado a una comunidad.	Párroco lo solicita	Parroquia San Clemente		Media	1,00	AC	16/10/2013	A	M	Firma de acta de pruebas del módulo Reporte de Asignación	Desarrollar un sistema consistente.	Cumplir con el alcance del proyecto.	1.3 CONSTRUCCION		No aplica. Probar interfaz de Personal Asignado Cumplir con el desarrollo del sistema

Tabla 23 Matriz de Trazabilidad de Requerimientos

## B. Estructura de la descomposición de trabajo WBS







### C. Informe de estado externo

<b>Fecha / Hora:</b>	Miércoles 31 de Julio del 2013 / 6:20 p.m.		
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco		
<b>Elaborado por</b>	Juan Castillo		
<b>Gerencia/Área</b>	GERENCIA DEL PROYECTO		
<b>Período</b>	Del 03 de Julio del 2013	Al 31 de Julio del 2013	
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO		

<b>PAQUETE DE TRABAJO</b>	Status del mes de Julio del 2013			
	<b>Actividad</b>	<b>Avance (%)</b>	<b>Fecha Fin</b>	<b>Observaciones</b>
Implementar Base de Datos en el gestor	Implementar Base de Datos en MySql	100%	05/07/2013	
Interfaces del Sistema	Diseñar interfaces del sistema	100%	10/07/2013	Diseño tentativo del sistema
Sub-Sistema Control de Sacramentos	Codificar Módulo de Feligrés	100%	17/07/2013	
	Ejecutar Pruebas del Módulo	100%	18/07/2013	
	Elaborar Acta de Pruebas	100%	19/07/2013	
	Codificar Módulo de Sacramentos	45%	19/08/2013	

<b>Fecha / Hora:</b>	Viernes 16 de Agosto del 2013 / 5:15 p.m.		
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco		
<b>Elaborado por</b>	Juan Castillo		
<b>Gerencia/Área</b>	GERENCIA DEL PROYECTO		
<b>Período</b>	Del 01 de Agosto del 2013	Al 16 de Agosto del 2013	
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO		

PAQUETE DE TRABAJO	Status del mes de Julio del 2013			
	Actividad	Avance (%)	Fecha Fin	Observaciones
Sub-Sistema Control de Sacramentos	Codificar Módulo de Sacramentos	95%	19/08/2013	

<b>Fecha / Hora:</b>	Viernes 30 de Agosto del 2013 / 5:30 p.m.			
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco			
<b>Elaborado por</b>	Juan Castillo			
<b>Gerencia/Área</b>	GERENCIA DEL PROYECTO			
<b>Período</b>	Del 17 de Agosto del 2013		Al 30 de Agosto del 2013	
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO			
PAQUETE DE TRABAJO	Status del mes de Julio del 2013			
	Actividad	Avance (%)	Fecha Fin	Observaciones
Sub-Sistema Control de Sacramentos	Codificar Módulo de Sacramentos	100%	19/08/2013	
	Ejecutar Pruebas del Módulo	100%	20/08/2013	
	Elaborar Acta de Pruebas	100%	28/08/2013	Entregado junto con el módulo de reportes de sacramentos
	Codificar Módulo de Reportes de Sacramentos	100%	26/09/2013	Adelanto de fechas de entrega
	Ejecutar Pruebas del Módulo	100%	27/08/2013	Adelanto de fechas de entrega
	Elaborar Acta de Pruebas	100%	28/08/2013	Adelanto de fechas de entrega

<b>Fecha / Hora:</b>	Viernes 13 de Septiembre del 2013 / 5:20 p.m.			
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco			
<b>Elaborado por</b>	Juan Castillo			
<b>Gerencia/Área</b>	GERENCIA DEL PROYECTO			
<b>Período</b>	Del 31 de Agosto del 2013		Al 13 de Septiembre del 2013	
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO			
<b>PAQUETE DE TRABAJO</b>	<b>Status del mes de Julio del 2013</b>			
	<b>Actividad</b>	<b>Avance (%)</b>	<b>Fecha Fin</b>	<b>Observaciones</b>
Sub-Sistema Control de Sacramentos	Codificar Módulo de Reportes de Sacramentos	100%	06/09/2013	
	Ejecutar Pruebas del Módulo	100%	10/09/2013	
	Elaborar Acta de Pruebas	100%	11/09/2013	
Sub-Sistema Asignación de Personal	Codificar Módulo de Personal	25%	18/09/2013	

<b>Fecha / Hora:</b>	Viernes 27 de Septiembre del 2013 / 5:20 p.m.		
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco		
<b>Elaborado por</b>	Juan Castillo		
<b>Gerencia/Área</b>	GERENCIA DEL PROYECTO		
<b>Período</b>	Del 14 de Septiembre del 2013		Al 27 de Septiembre del 2013
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO		
<b>PAQUETE DE TRABAJO</b>	<b>Status del mes de Julio del 2013</b>		
	<b>Actividad</b>	<b>Avance (%)</b>	<b>Fecha Fin</b>

Sub-Sistema Asignación de Personal	Codificar Módulo de Personal	100%	18/09/2013	
	Ejecutar Pruebas del Módulo	100%	20/09/2013	
	Elaborar Acta de Pruebas	100%	23/09/2013	
	Codificar Módulo de Comunidades	65%	30/09/2013	

<b>Fecha / Hora:</b>	Viernes 11 de Octubre del 2013 / 5:20 p.m.						
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco						
<b>Elaborado por</b>	Juan Castillo						
<b>Gerencia/Área</b>	GERENCIA DEL PROYECTO						
<b>Período</b>	Del 28 de Septiembre del 2013	Al 11 de Octubre del 2013					
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO						
<b>PAQUETE DE TRABAJO</b>	Status del mes de Julio del 2013						
	<b>Actividad</b>	<b>Avance (%)</b>	<b>Fecha Fin</b>	<b>Observaciones</b>			
Sub-Sistema Asignación de Personal	Codificar Módulo de Comunidades	100%	30/09/2013				
	Ejecutar Pruebas del Módulo	100%	03/10/2013				
	Elaborar Acta de Pruebas	100%	04/10/2013				
	Codificar Módulo de Reportes de Asignación	95%	14/10/2013				

<b>Fecha / Hora:</b>	Viernes 25 de Octubre del 2013 / 5:20 p.m.						
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco						
<b>Elaborado por</b>	Juan Castillo						
<b>Gerencia/Área</b>	GERENCIA PROYECTO						
<b>Período</b>	Del 12 de Octubre del 2013	Al 25 de Octubre del 2013					
<b>Referencia</b>	PLANTILLA DE ACTA DE REPORTE DE RENDIMIENTO						
<b>PAQUETE DE TRABAJO</b>	Status del mes de Julio del 2013						
	<b>Actividad</b>	<b>Avance (%)</b>	<b>Fecha Fin</b>	<b>Observaciones</b>			
Sub-Sistema Asignación de Personal	Codificar Módulo de Reportes de Asignación	100%	14/10/2013				
	Ejecutar Pruebas del Módulo	100%	15/10/2013				
	Elaborar Acta de Pruebas	100%	14/10/2013				
Backups	Codificar Módulo de Backups	100%	23/10/2013				
	Ejecutar Pruebas del Módulo	100%	24/10/2013				
	Elaborar Acta de Pruebas	100%	14/10/2013				

## D. Riesgos del proyecto

ID	Descripción	Acciones de Prevención	Acciones de Corrección
R01	Requisitos poco claros	Realización de varias reuniones con el cliente; elaboración de cuestionarios para aclarar puntos poco claros de las reuniones previas.	En las primeras fases se realizarán los cambios necesarios para incorporar los nuevos requisitos o los cambios necesarios para que se cumpla con la funcionalidad solicitada. En las fases de Construcción y Transición se valorará la importancia de las modificaciones/requisitos nuevos frente a la cantidad de tiempo disponible para abordarlos. En caso de que se decida aceptarlos, se revisarán los requisitos afectados, así como toda la documentación y código derivado de los mismos hasta el punto de aparición del cambio.
R04	Diseño Erróneo	Durante la fase de Elaboración se desarrollará en paralelo un prototipo conteniendo la arquitectura del sistema para comprobar la validez de la misma. En caso de encontrarse errores o inconsistencias, podrá modificarse el diseño al mismo tiempo que la implementación del prototipo.	Si el riesgo se convierte en hecho durante la fase de Elaboración, se revisará y modificará la documentación de diseño afectada. Si lo hace durante la fase de construcción, se estudiará una solución acorde a los tiempos de plazo de que se dispone. La planificación se reajustará si fuera necesario.
R05	Falta de un Experto	Aprendizaje continuo durante todo el proyecto	Las dudas que no se sepan resolver se trasladarán al tutor y a foros especializados.
R09	Falta de seguimiento permanente de tareas y actividades	Llevar al día una revisión del estado del proyecto para anotar los posibles atrasos y poder así tomar medidas en el instante.	Realizar una recandelerización de tareas, así como llamadas de atención a los miembros del equipo que dejen sus tareas para última instancia.

Tabla 24 Tabla de Riesgos del Proyecto

## E. Solicitud de cambios

### Petición de Cambio al Cronograma

#### Datos del Solicitante

Código de Petición :	Pet-001	
Fecha de Solicitud:	19/09/2013	
Nombre Solicitante	Vicente Diez Varona	
Área Promotora		
Cargo solicitante Párroco	Teléfono (056)532148	E-Mail

#### Datos de la Petición de Cambio al Cronograma

Título :	Petición para adelantar la fecha de fin del proyecto
Prioridad:	Máxima
Descripción Simplificada de la Petición	Se pide cambiar la fecha de fin del proyecto del 06 de Diciembre del 2013 inicial a la fecha 21 de Noviembre del 2013.
Descripción del Motivo	El motivo es ajustar el tiempo del desarrollo del proyecto a fechas anteriores del mes de Diciembre y de fines de Noviembre, debido a que en el último mes del año se realizan diversos preparativos para festividades como son la misa de Navidad y la misa de Año Nuevo entre otros. Habiendo analizado los tiempos se pide que se adelante la fecha de fin del proyecto para que no se cruce con los preparativos que se tendrán.
Documentos Adjuntos	

#### Del Cambio al Cronograma

Objetivo :	Adelantar la fecha de fin del proyecto del 06 de Diciembre del 2013 inicial, al 21 de Noviembre del 2013.
Alcance:	Cronograma de trabajo y las actividades que se ejecutan a partir de la fecha de la petición. Documentación de la fase de iniciación y planificación donde se contempla la fecha de fin del proyecto.

#### Riesgos involucrados de no implementar el Cambio al Cronograma

Ítem :	Descripción del Riesgo	Responsable del Riesgo	Impacto en el Proyecto (A, M, B)

01	Alta probabilidad del retraso de aprobación de los documentos del cierre del proyecto, debido al cruce de fechas del fin del proyecto con los preparativos de las actividades del mes de diciembre del 2013.	Gerente del Proyecto	Medio
----	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------	-------



Sponsor

Vicente Diez Varona

### Petición de Cambio a los Requerimientos

#### Datos del Solicitante

Código de Petición :	Pet-002	
Fecha de Solicitud:	20/09/2013	
Nombre Solicitante	Vicente Diez Varona	
Área Promotora		
Cargo solicitante Párroco	Teléfono (056)532148	E-Mail

#### Datos de la Petición de Cambio / Nueva Funcionalidad

Título :	Petición para el cambio de los requerimientos del proyecto
Prioridad:	Máxima
Descripción Simplificada de la Petición	Se pide realizar un cambio a los requerimientos iniciales del proyecto para la eliminación de la función "Eventos" del sistema.
Descripción del Motivo	El motivo es el resultado de un análisis de las funciones propuestas del sistema resultante del proyecto, llegando a la conclusión que una de las funciones del sistema, el cuál es el registro del personal a los eventos realizados, no es una necesidad prioritaria o de alta preocupación para el estado actual de la parroquia, ya que el personal está siempre en constante formación no sólo de los eventos de la parroquia sino de las propias actividades de sus respectivas comunidades.

<b>Documentos Adjuntos</b>	
--------------------------------	--

### **Del Cambio o Nuevo Requerimiento Propuesto**

<b>Objetivo :</b>	Eliminar los requerimientos del proyecto que contemplen la función "Eventos" del sistema
<b>Alcance:</b>	Documentación de la fase de iniciación y planificación donde se contempla este requerimiento y/o función del sistema.

### **Mejoras o Beneficios Esperados**

<b>Ítem :</b>	<b>Descripción de la Mejora o Beneficio</b>	<b>Importancia (alta, media, baja)</b>
1	Obtener un sistema que cubra las necesidades prioritarias de la parroquia.	Alta
2	Aumentar el tiempo de pruebas y corrección de errores del sistema.	Alta



Sponsor

Vicente Diez Varona

## 4.2. Ingeniería del proyecto

### A. Concepción (Incepción)

#### Especificación de requerimientos de software

Involucrado	Cod. Req	Descripción del requerimiento	Categoría	Prioridad	Criterio de aceptación
Secretaria	Req-001	Actualizar los datos de los países. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de países, durante las pruebas de Control de Calidad
Secretaria	Req-002	Actualizar los datos de las ciudades o departamentos. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de las ciudades, durante las pruebas de Control de Calidad
Secretaria	Req-003	Actualizar los datos de las provincias del Perú. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de las provincias, durante las pruebas de Control de Calidad
Secretaria	Req-004	Para los países, ciudades o departamento y provincias, sólo se podrán eliminar los nuevos registros que se ingresen en el sistema.	Req. Funcional	Media	Impedir la eliminación de los registros por defecto de los países, ciudades y provincias.
Secretaria	Req-005	Actualizar los datos de los templos: Nombre, tipo, lugar. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de los templos, durante las pruebas de Control de Calidad
Secretaria	Req-	Actualizar los datos del feligrés.	Req.	Alta	Cero errores al momento de

	006	Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Funcional		probar las funciones de la interfaz de los feligreses, durante las pruebas de Control de Calidad
Secretaria	Req-007	Posibilidad de registrar a un personal como feligrés y viceversa.	Req. Funcional	Media	Casilla de verificación en la interfaz de feligrés si se tiene los datos del personal. Casilla de verificación en la interfaz del personal si se tiene los datos del feligrés.
Secretaria	Req-008	Actualizar los datos del Bautismo: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Natural del Padre, Nombres de la Madre, Natural de la Madre, Lugar de nacimiento, Fecha de nacimiento, Fecha de Bautismo, Nombre del Padrino, Nombre de la Madrina, Ministro o Celebrante, Nota Marginal. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de bautismos, durante las pruebas de Control de Calidad
Secretaria	Req-009	Actualizar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios, Nº de registros por folio, fecha de apertura, fecha de cierre, estado, cantidad de registros. Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reaperturar Libro, Buscar y	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de los libros sacramentales, durante las pruebas de Control de Calidad

		Cancelar.			
Secretaria	Req-010	Permitir la reapertura de un libro sacramental si el total de registros no es igual a la cantidad máxima soportada.	Req. Funcional	Media	Habilitar la opción de reapertura del libro sacramental, si la cantidad máxima de registros permitidos es mayor a la cantidad de registros.
Secretaria	Req-011	Modificar los datos de un libro sacramental, sólo si está aperturado.	Req. Funcional	Media	Inhabilitar el control de actualización de datos del libro sacramental si está cerrado.
Secretaria	Req-012	Impedir el registro de sacramentos si no hay ningún libro de dicho sacramento aperturado.	Req. Funcional	Alta	Impedir el inicio de la interfaz del sacramento si no hay ningún libro sacramental de dicho sacramento abierto. Mostrar mensaje al usuario.
Secretaria	Req-013	Actualizar los datos específicos de la Confirmación: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Nombres de la Madre, Lugar de nacimiento, Fecha de nacimiento, Lugar de Bautismo, Fecha de Bautismo, Nº Libro – Fojas – Número del Bautismo, Fecha de la Confirmación, Nombre del Padrino o Madrina, Ministro o Celebrante, Anotaciones. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de confirmaciones, durante las pruebas de Control de Calidad

Secretaria	Req-014	Actualizar los datos específicos del Matrimonio: Nombre de la Parroquia, Nº Libro – Fojas – Número, Nombres del Esposo, Nacido en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Nombres de la Esposa, Nacida en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Fecha de Bautismo, Fecha del Matrimonio, Nombre del Padrino, Nombre de la Madrina, Nombre de los Testigos (2), Ministro o Celebrante, Anotaciones. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de matrimonios, durante las pruebas de Control de Calidad
Secretaria	Req-015	Posibilidad de ingresar los datos del bautismo del feligrés en la confirmación (esposo o esposa en el matrimonio) manualmente, en caso que el feligrés no tenga registros de bautismo en la parroquia.	Req. Funcional	Alta	Habilitación automática de los campos correspondientes en caso de no encontrar el registro.
Secretaria	Req-016	No permitir la modificación de los registros de sacramentos para los campos: feligrés y libro, folio, número.	Req. Funcional	Alta	Inhabilitar los campos: feligrés, libro, folio y número, durante el proceso de actualización de datos de los sacramentos.
Secretaria	Req-017	Ordenar los registros del feligrés y personal alfabéticamente por apellidos.	Req. No Funcional	Media	Ordenación de manera correcta por el apellido del feligrés al momento de realizar la búsqueda.
Secretaria	Req-	Buscar los sacramentos por el	Req. No	Media	Contemplar el apellido

	018	Apellido Paterno del feligrés.	Funcional		paterno como filtro de búsqueda en los sacramentos
Párroco	Req-019	Actualizar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz del personal, durante las pruebas de Control de Calidad
Párroco	Req-020	Actualizar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de las comunidades, durante las pruebas de Control de Calidad
Párroco	Req-021	Actualizar los datos de tipos de personal y de comunidades. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Media	Cero errores al momento de probar las funciones de la interfaz de tipos de personal y comunidades, durante las pruebas de Control de Calidad
Párroco	Req-022	Actualizar los datos de los lugares de reunión de las comunidades Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de lugares de reunión, durante las pruebas de Control de Calidad
Párroco	Req-023	Posibilidad de añadir un personal a distintas comunidades y eliminar una	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la

		asignación de un personal.			interfaz de asignaciones, durante las pruebas de Control de Calidad
Secretaria	Req-024	Buscar los registros de forma rápida y precisa.	Req. No Funcional	Media	Función correcta de los filtros de búsqueda
Párroco	Req-025	Realizar copias de seguridad de la base de datos del sistema.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de backup's, durante las pruebas de Control de Calidad
Párroco	Req-026	Restablecer las copias de seguridad generadas del sistema.	Req. Funcional	Alta	Cero errores al momento de probar las funciones de la interfaz de restauración, durante las pruebas de Control de Calidad
Secretaria	Req-027	Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros soportados.	Req. No Funcional	Media	Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros permitidos. Mostrar un mensaje al usuario.
Párroco	Req-028	Imprimir los datos de los sacramentos con el formato de actas de la parroquia	Req. No Funcional	Alta	Cero errores al momento de generar el pdf de las actas de sacramentos
Párroco	Req-029	Imprimir la lista de personal asignado a una comunidad	Req. No Funcional	Alta	Cero errores al momento de generar el pdf del personal asignado

Tabla 25 Especificación de Requerimientos de Software

### Modelos de caso de uso del sistema

- Caso de Uso de Feligrés

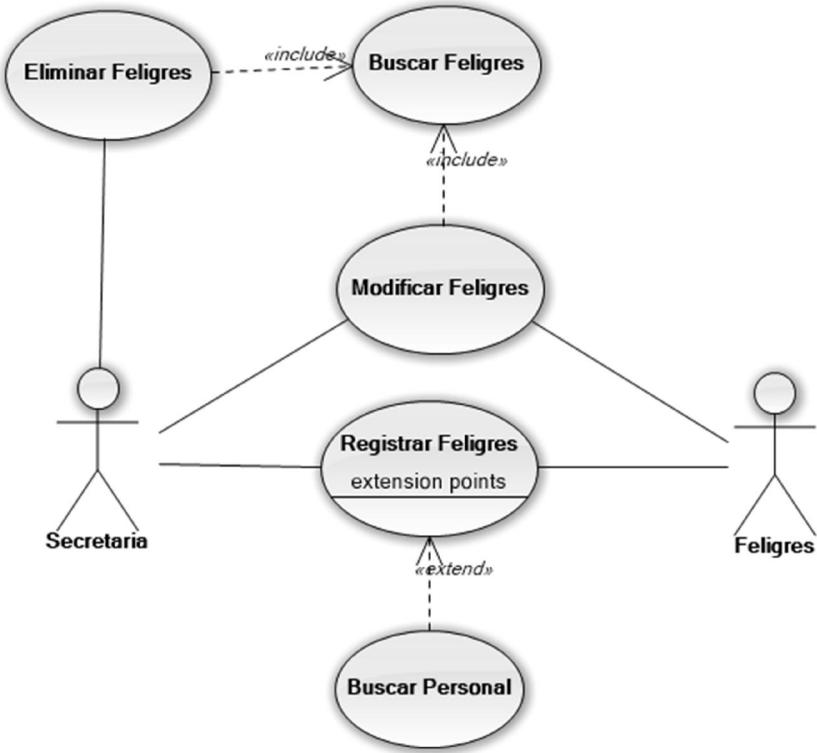


Ilustración 10 Caso de Uso de Feligrés

- Caso de Uso de Templos

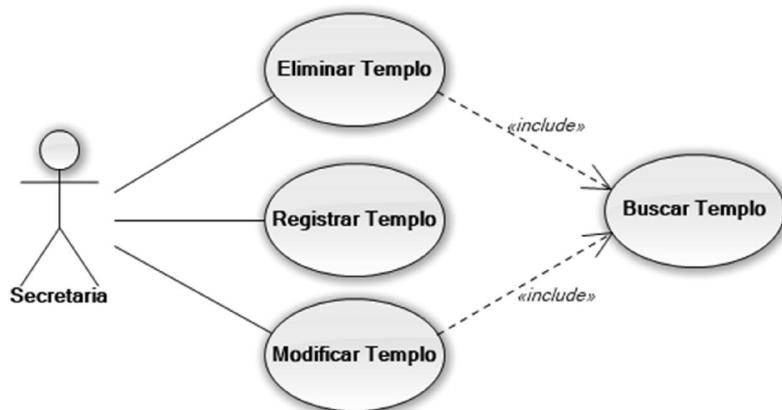


Ilustración 11 Caso de Uso de Templos

- Caso de Uso del Libro Sacramental

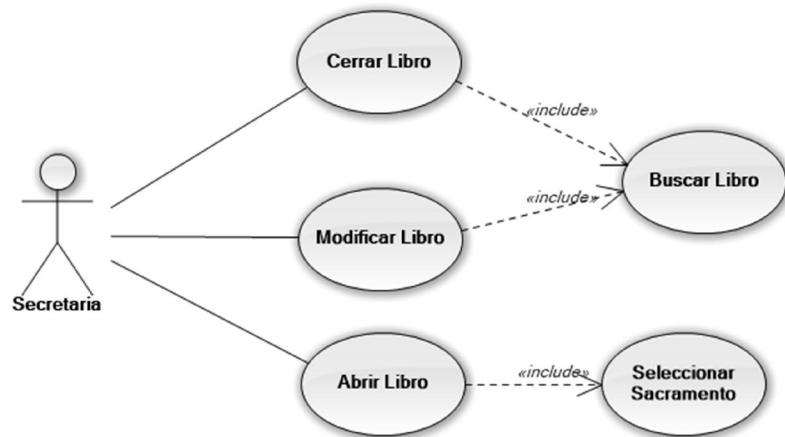


Ilustración 12 Caso de uso del Libro Sacramental

- Caso de Uso de Sacramentos

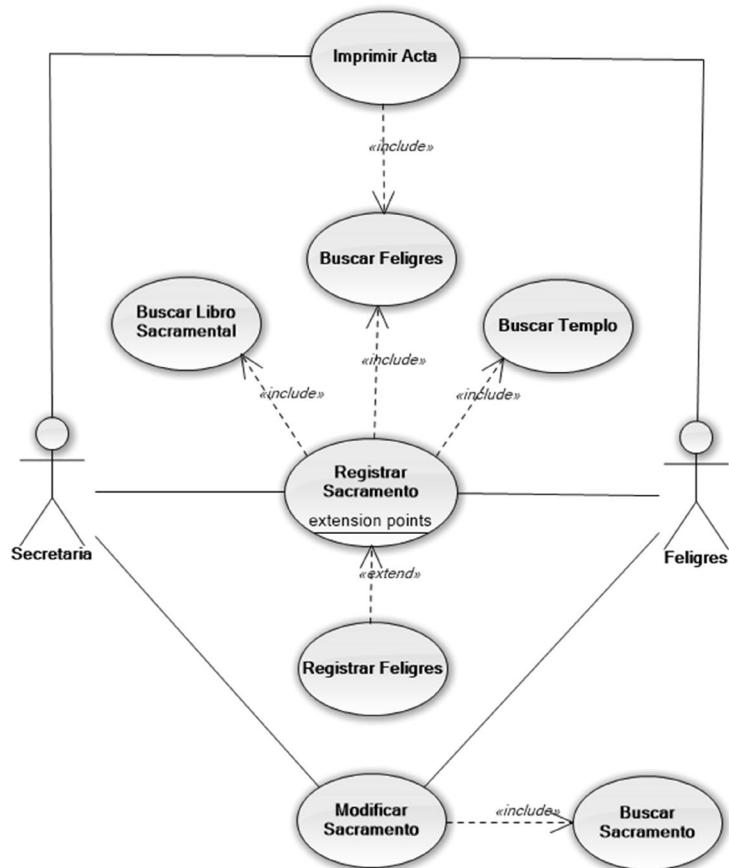


Ilustración 13 Caso de Uso de Sacramentos

- Caso de Uso de Personal

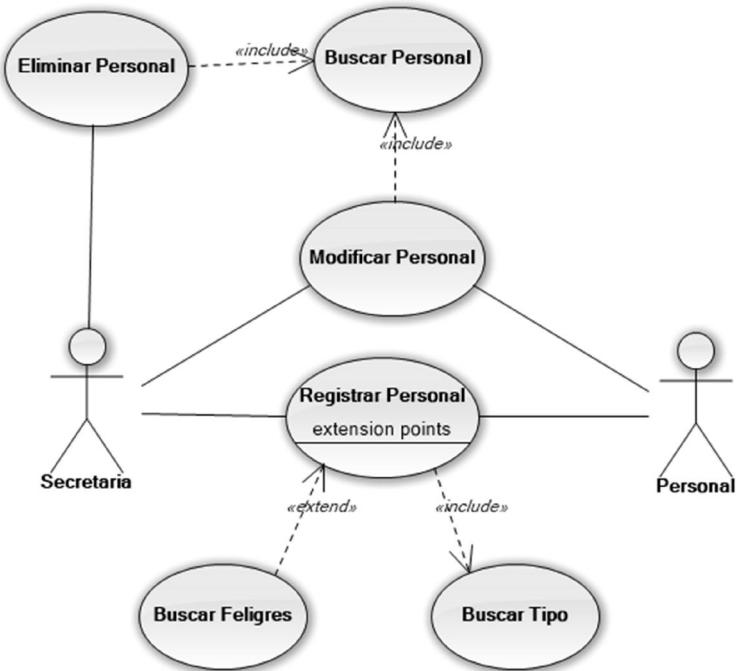


Ilustración 14 Caso de Uso de Personal

- Caso de Uso de Comunidades

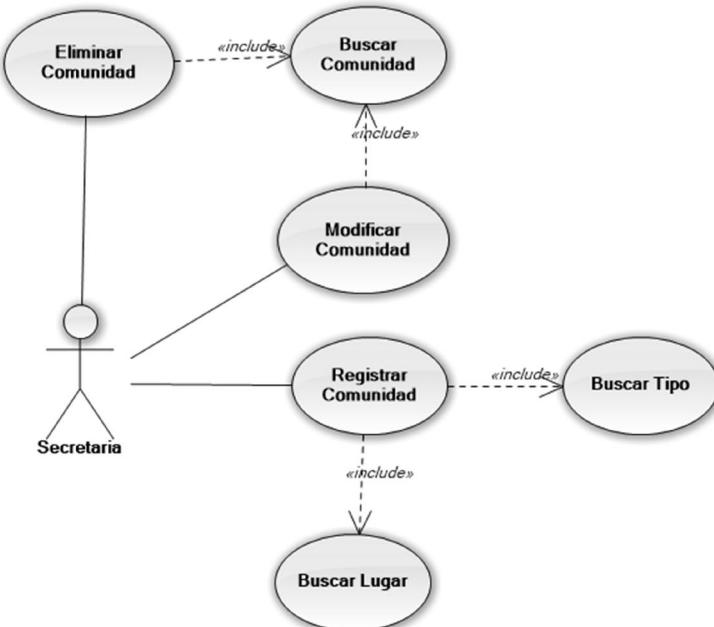


Ilustración 15 Caso de Uso de Comunidades

- Caso de Uso de Asignación

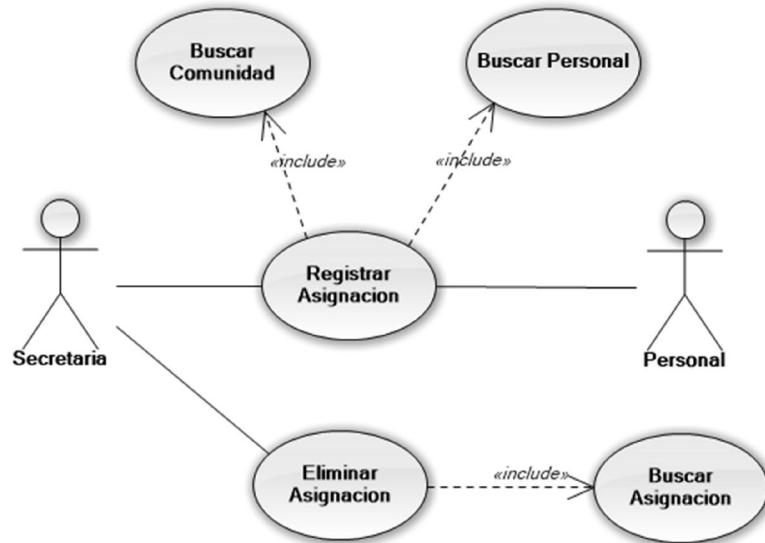


Ilustración 16 Caso de Uso de Asignación

Diagrama de paquetes

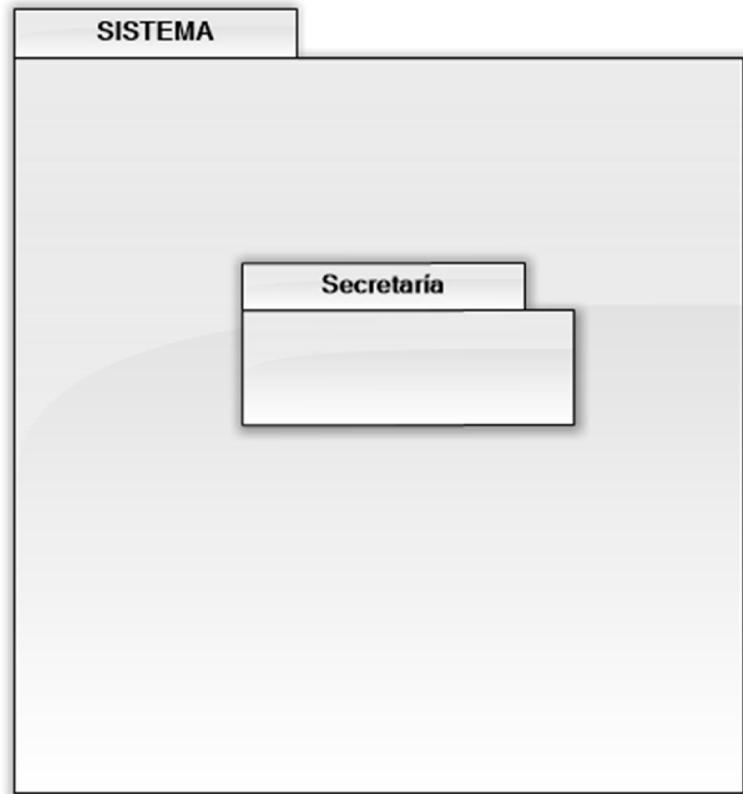


Ilustración 17 Diagrama de Paquetes

## B. Elaboración

### Diseño del sistema y arquitectura del software

Prototipo del menú Básico



Ilustración 18 Prototipo del menú Básico

Prototipo del menú Control de Sacramentos



Ilustración 19 Prototipo del menú Control de Sacramentos

Prototipo del menú Asignación de Personal

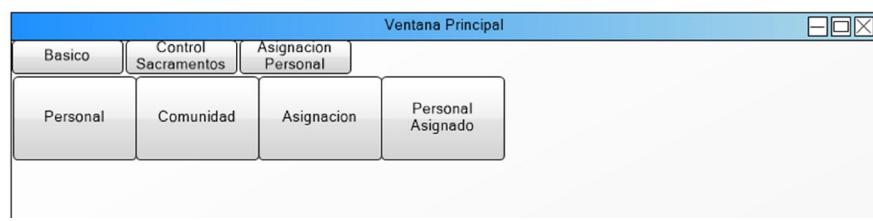


Ilustración 20 Prototipo del menú Asignación de Personal

### Prototipo de la interfaz de Países



Ilustración 21 Prototipo de la interfaz de Países

### Prototipo de la interfaz de Ciudades

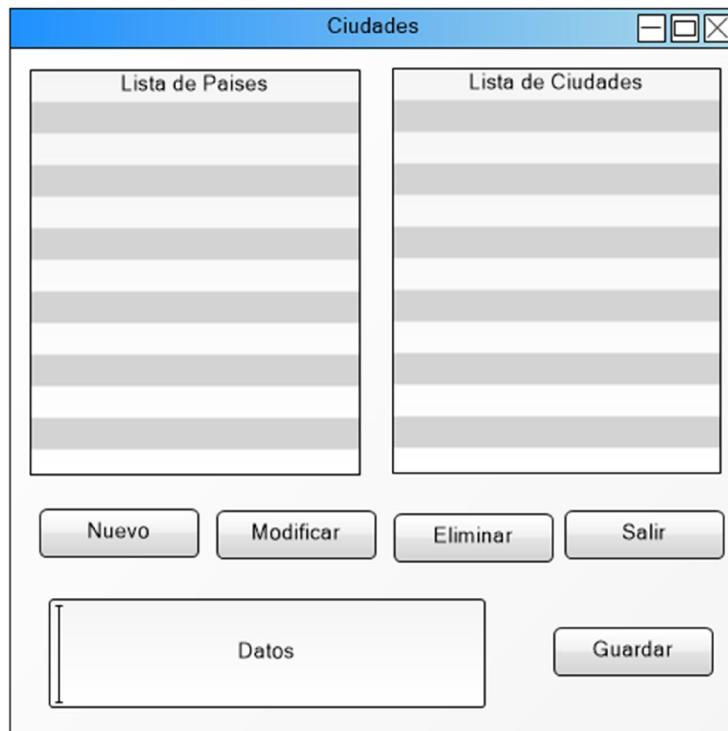


Ilustración 22 Prototipo de la interfaz de Ciudades

### Prototipo de la interfaz de Ciudades

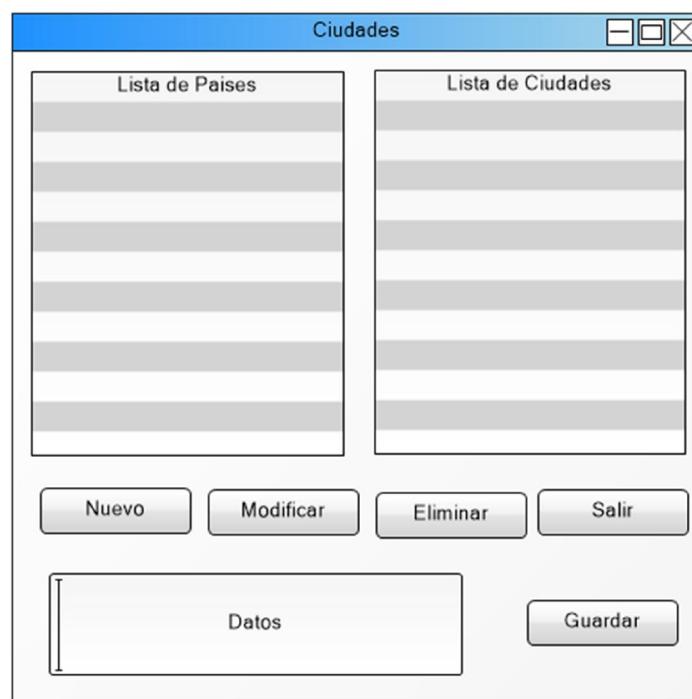


Ilustración 23 Prototipo de la interfaz de Ciudades

### Prototipo de la interfaz de Tipos

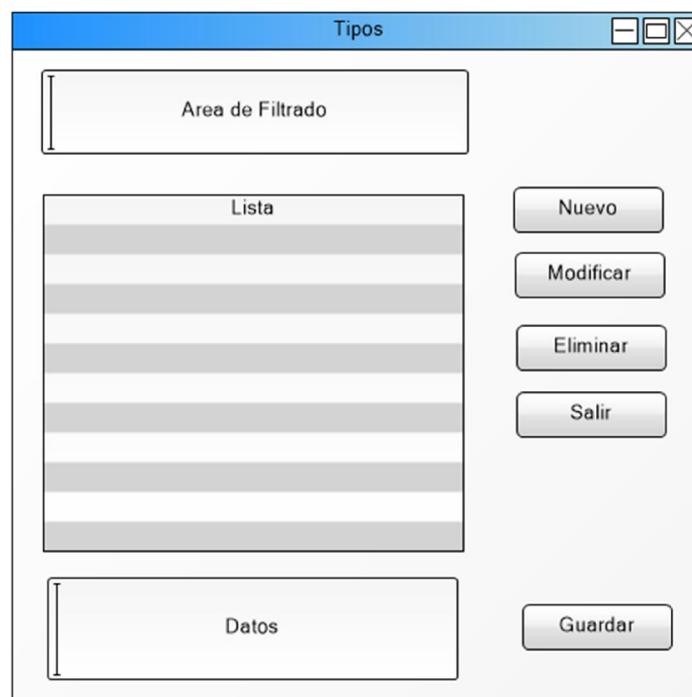


Ilustración 24 Prototipo de la interfaz de Tipos

### Prototipo de la interfaz de Lugares de Reunión



Ilustración 25 Prototipo de la interfaz de Lugares de Reunión

### Prototipo de la interfaz de Salvar Datos

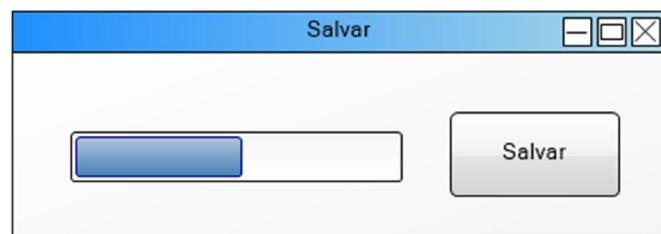


Ilustración 26 Prototipo de la interfaz de Salvar Datos

### Prototipo de la interfaz de Restaurar Datos



Ilustración 27 Prototipo de la interfaz de Restaurar Datos

## Prototipo de la interfaz de Feligrés

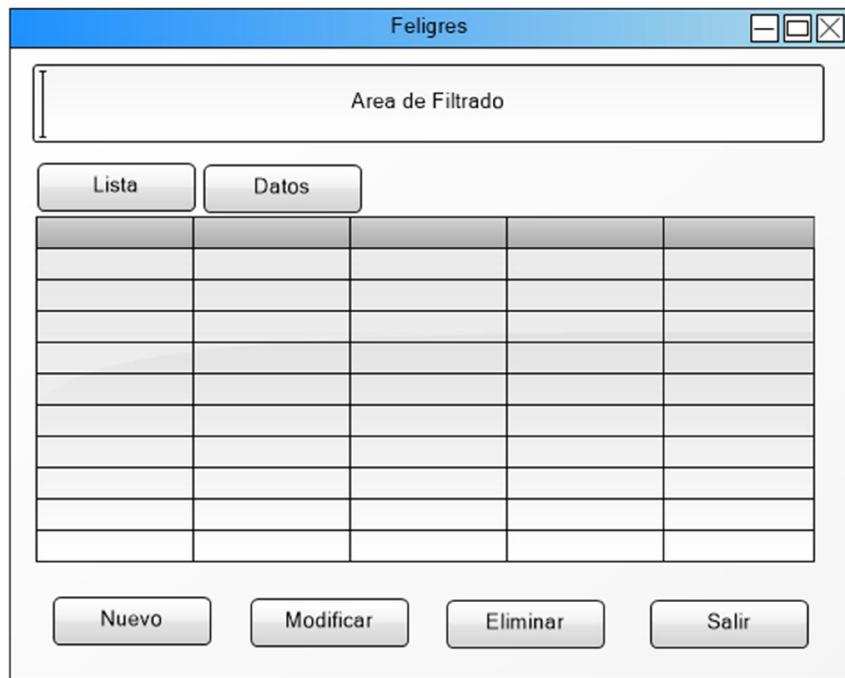


Ilustración 28 Prototipo de la interfaz de Feligrés 1

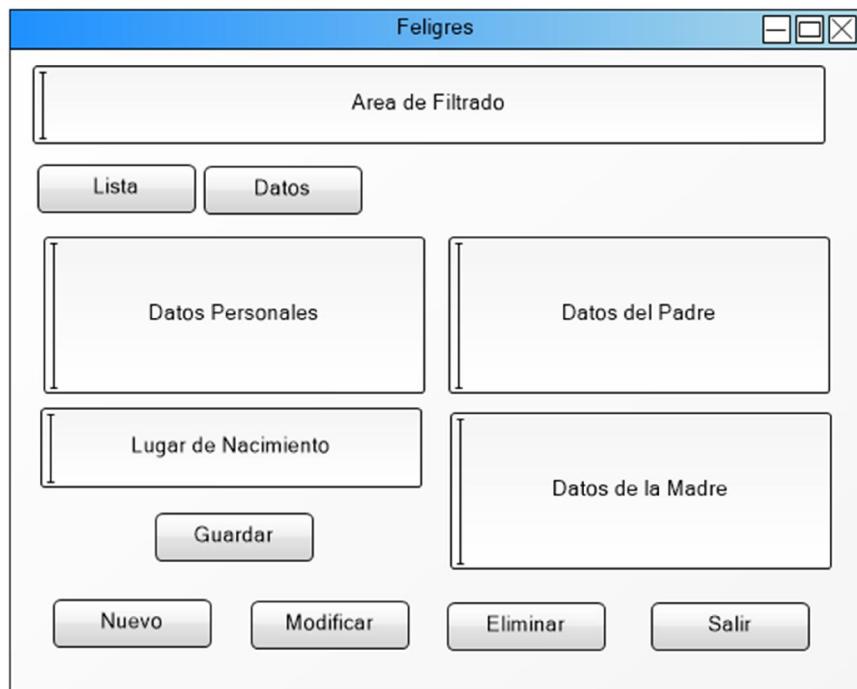


Ilustración 29 Prototipo de la interfaz de Feligrés 2

### Prototipo de la interfaz del Libro Sacramental



Ilustración 30 Prototipo de la interfaz del Libro Sacramental

### Prototipo de la interfaz de Templos



Ilustración 31 Prototipo de la interfaz de Templos

## Prototipo de la interfaz de Bautismos



Ilustración 32 Prototipo de la interfaz de Bautismos 1

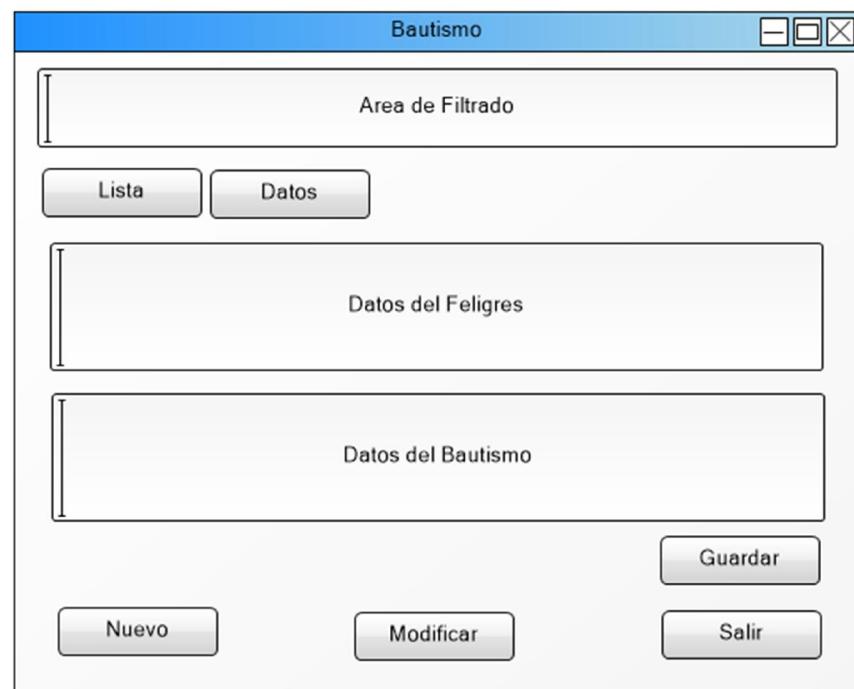


Ilustración 33 Prototipo de la interfaz de Bautismos 2

## Prototipo de la interfaz de Confirmaciones

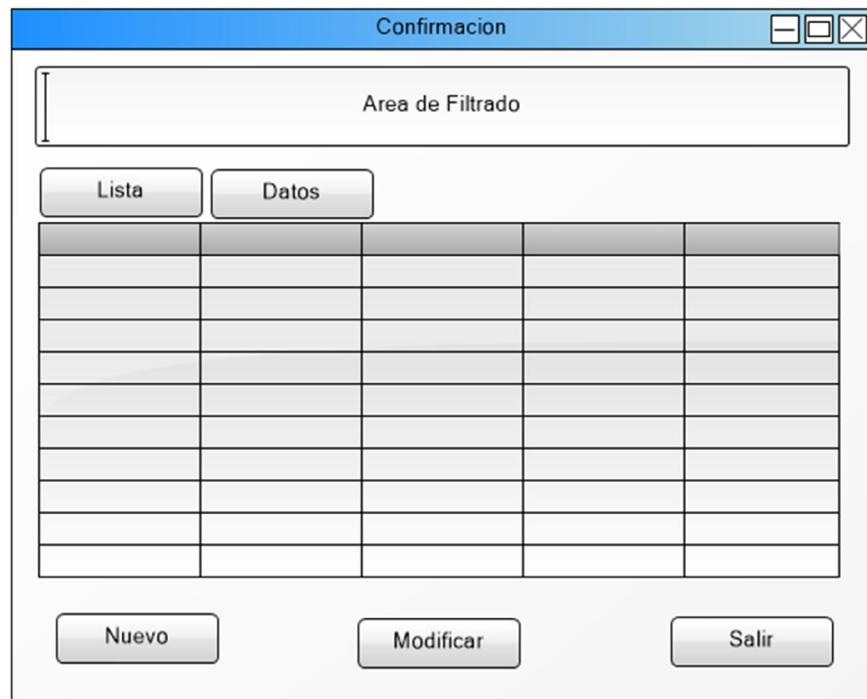


Ilustración 34 Prototipo de la interfaz de Confirmaciones 1

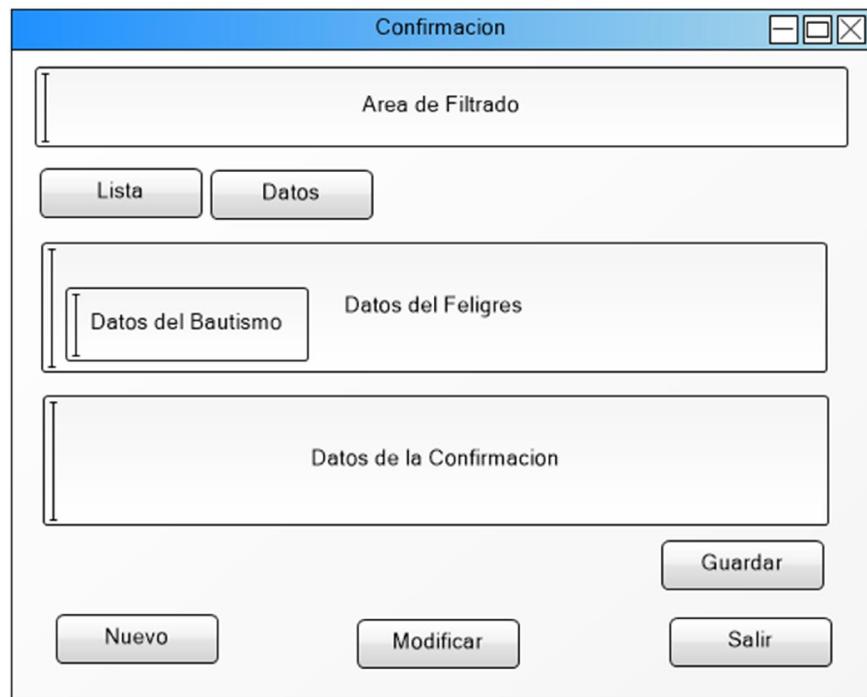


Ilustración 35 Prototipo de la interfaz de Confirmaciones 2

## Prototipo de la interfaz de Matrimonios



Ilustración 36 Prototipo de la interfaz de Matrimonios 1



Ilustración 37 Prototipo de la interfaz de Matrimonios 2



Ilustración 38 Prototipo de la interfaz de Matrimonios 3

#### Prototipo de la interfaz de Actas de Sacramentos



Ilustración 39 Prototipo de la interfaz de Actas de Sacramentos

## Prototipo de la interfaz de Personal



Ilustración 40 Prototipo de la interfaz de Personal 1

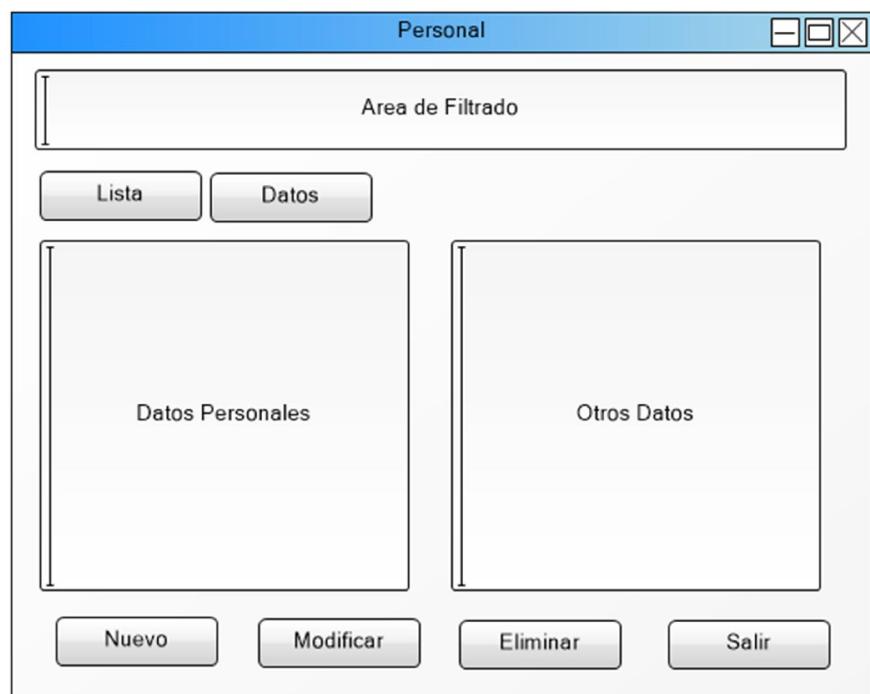


Ilustración 41 Prototipo de la interfaz de Personal 2

### Prototipo de la interfaz de Comunidades



Ilustración 42 Prototipo de la interfaz de Comunidades

### Prototipo de la interfaz de Asignaciones

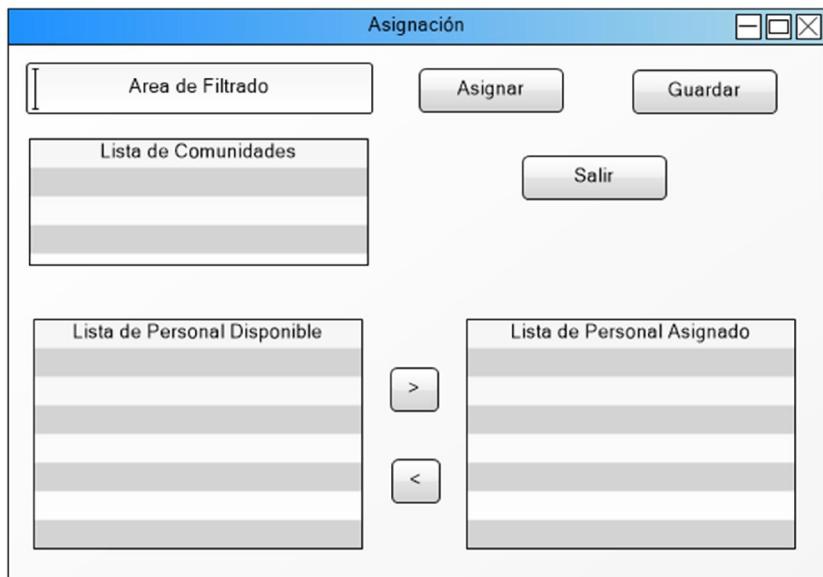


Ilustración 43 Prototipo de la interfaz de Asignaciones

## Prototipo de la interfaz de Personal Asignado

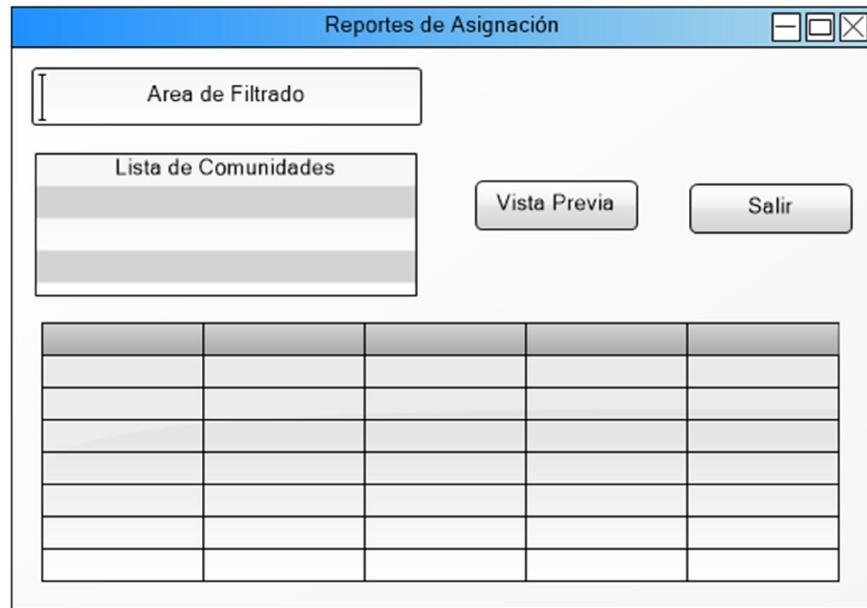


Ilustración 44 Prototipo de la interfaz de Personal Asignado

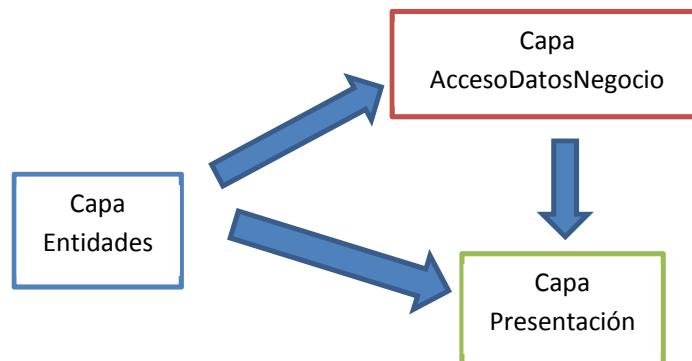
### Arquitectura

Para la construcción del Sistema se utilizó la arquitectura N-Capas, para poder tener un mayor control al momento de codificar el sistema y realizar correcciones del código y funcionamiento del mismo.

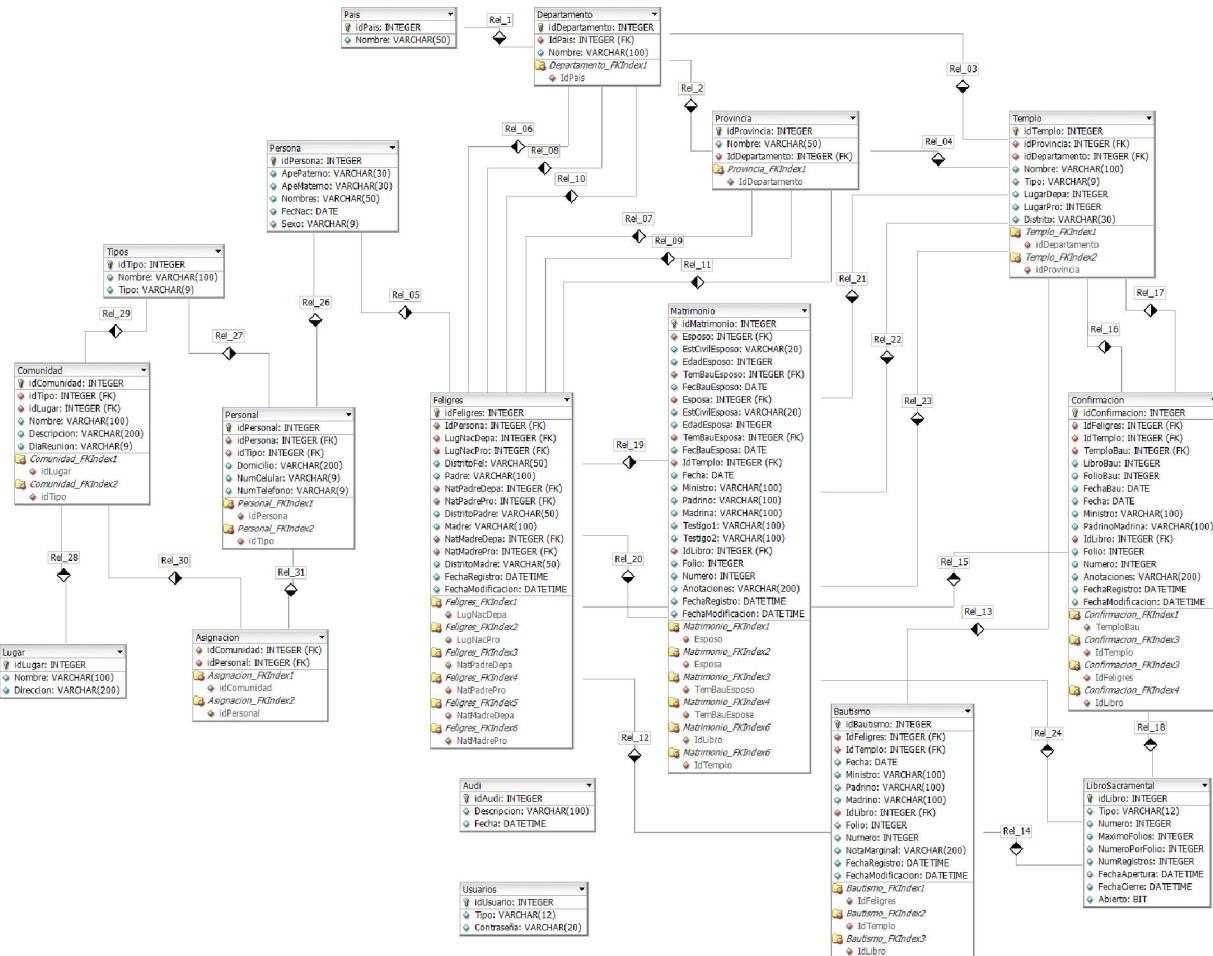
Las capas utilizadas para el desarrollo son:

- Capa Entidades => Donde se encuentran las clases que servirán para controlar los datos del Sistema. Cada clase hace referencia a una tabla de la Base de Datos.
- Capa AccesoDatosNegocio => Donde se encuentran los métodos que conectan a la Base de Datos con el Sistema, y poder realizar las distintas funciones tales como: registrar, modificar, eliminar, etc... También se está incluyendo la Lógica que siguen estos métodos para su uso.
- Capa Presentación => Donde se encuentra el diseño del sistema y con el que el usuario interactuará para controlar los datos del sistema.

Las Capas están relacionadas de la siguiente manera:



## Modelo de datos



## Ilustración 45 Diagrama de Clases

## C. Construcción

### Entorno de construcción, desarrollo y pruebas preparado

El desarrollo del sistema se lleva a cabo utilizando el lenguaje de programación C# haciendo uso del IDE Visual Studio 2012 Express for Desktop.

Se hace uso de un solo proyecto para vincular toda la arquitectura de N-Capas correspondiente al desarrollo del sistema, así como poder integrar las librerías auxiliares que brindan apoyo al código.

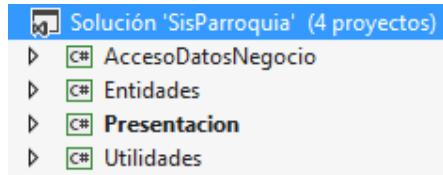


Ilustración 46 Capas del Sistema

NOTA: El último proyecto “Utilidades” es una librería donde se encuentra el código que brinda apoyo a las capas. Contiene código que se repite en la mayoría de las interfaces.

Los elementos que contendrán cada Capa se definen a continuación:

- Capa Entidades
  - Clases que refieren a las tablas de la Base de Datos, con sus respectivos campos, para permitir un adecuado control de los datos.

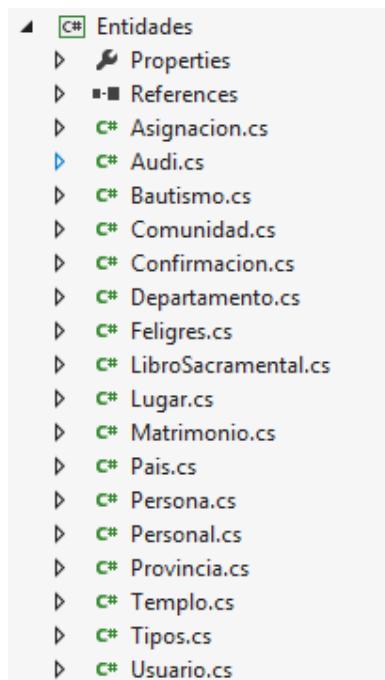


Ilustración 47 Estructura de la Capa Entidades

- Capa AccesoDatosNegocio
  - Manejo de la conexión con la Base de Datos
  - Métodos de conexión con los procedimientos almacenados de la Base de Datos, por cada Tabla. Estos son privados.
  - Métodos que contienen la Lógica de Negocio que seguirán los métodos privados para ser ejecutados. Estos son públicos.

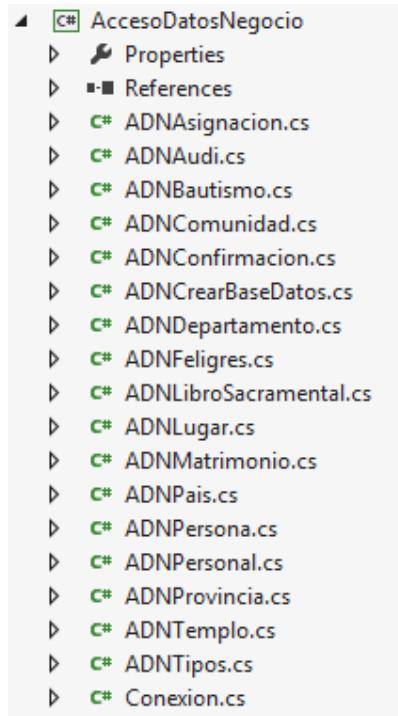


Ilustración 48 Estructura de la Capa AccesoDatosNegocio

- Capa Presentación
  - Ventanas o interfaces, con las cuáles el usuario podrá interactuar con la Base de Datos.

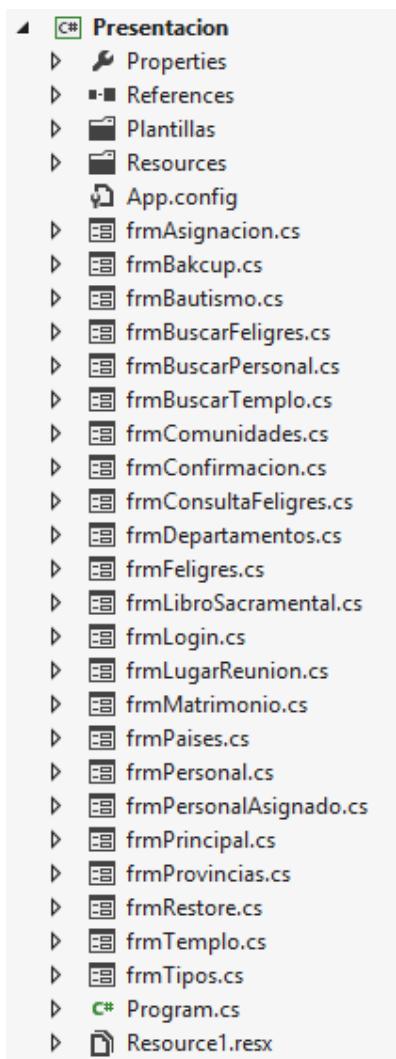


Ilustración 49 Estructura de la Capa Presentación

La relación de los componentes se lleva a cabo de la siguiente manera:

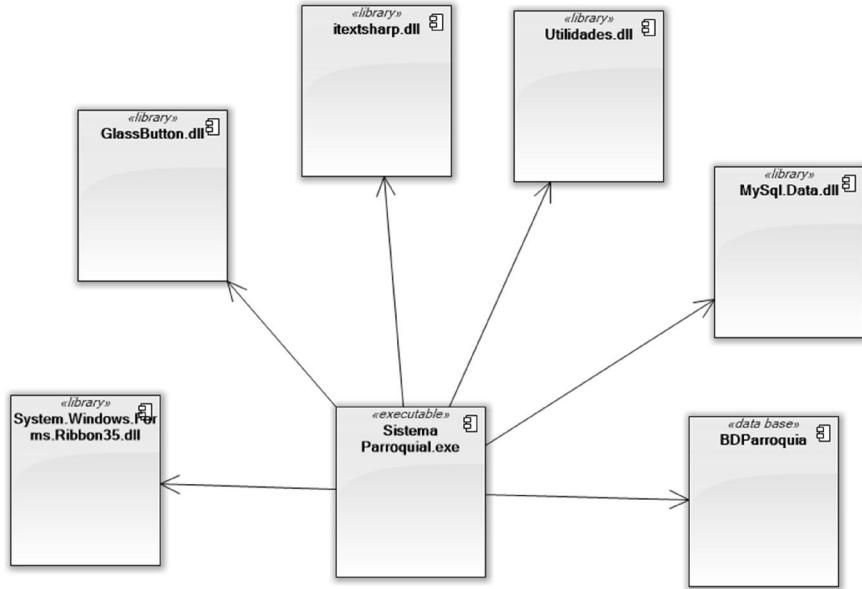


Ilustración 50 Diagrama de Componentes

En cuanto a las pruebas, se llevarán a cabo utilizando los formatos de pruebas unitarias expuestos anteriormente.

El paso a paso de las pruebas se apoyará en los Casos de Prueba de los Casos de Uso, los cuales se definen en el siguiente punto, luego del código fuente.

### Construcción de software y pruebas unitarias

#### **Código fuente de la capa Entidades**

- Entidad Asignación

```
namespace Entidades
{
    public class Asignacion
    {
        public int IdComunidad { get; set; }
        public int IdPersonal { get; set; }

        public Asignacion() { }

        public Asignacion(int IdComunidad, int IdPersonal)
        {
            this.IdComunidad = IdComunidad; this.IdPersonal
                = IdPersonal;
        }
    }
}
```

- Entidad Audi

```
namespace Entidades
{
    public class Audi
```

```

    {
        public string descripcion { get; set; }

        public Audi() { }

        public Audi(string descripcion)
        {
            this.descripcion = descripcion;
        }
    }
}

```

- Entidad Bautismo

```

namespace Entidades
{
    public class Bautismo
    {
        public int IdBautismo { get; set; }
        public Feligres NomFeligres { get; set; }
        public Templo NomTemplo { get; set; }
        public DateTime Fecha { get; set; }
        public string Ministro { get; set; }
        public string Padrino { get; set; }
        public string Madrina { get; set; }
        public LibroSacramento Libro { get; set; }
        public int Folio { get; set; }
        public int Numero { get; set; }
        public string NotaMarginal { get; set; }

        public Bautismo() { }

        public string[] DatosToArray()
        {
            string[] aux = new string[16];
            aux[0]=IdBautismo.ToString();
            aux[1]=NomFeligres.IdFeligres.ToString();
            aux[2]=NomFeligres.NomPersona.ApePaterno+
                "+NomFeligres.NomPersona.ApeMaterno+
                ", " + NomFeligres.NomPersona.Nombres;
            aux[3] = NomTemplo.IdTemplo.ToString();
            aux[4] = NomTemplo.Nombre + " de " +
                NomTemplo.LugarPro.Nombre + " " +
                NomTemplo.Distrito;
            aux[5]=Fecha.ToString("dd/MM/yyyy");
            aux[6]=Ministro;
            aux[7]=Padrino;
            aux[8]=Madrina;
            aux[9]=Libro.IdLibro.ToString();
            aux[10]=Folio.ToString();
            aux[11]=Numero.ToString();
            aux[12]="Libro "+Libro.Numero+", folio
                "+Folio+", número "+Numero;
            aux[13]=NotaMarginal;
            aux[14] = Libro.Abrierto.ToString();
            return aux;
        }
    }
}

```

- Entidad Comunidad

```

namespace Entidades
{

```

```

public class Comunidad
{
    public int IdComunidad { get; set; }
    public string Nombre { get; set; }
    public string Descripcion { get; set; }
    public Tipos NomTipo { get; set; }
    public Lugar Lugar { get; set; }
    public string DiaReunion { get; set; }

    public Comunidad() { }

    public Comunidad(int IdComunidad) { this.IdComunidad = IdComunidad; }

    public string[] DatosToArray()
    {
        string[] aux = new string[7];
        aux[0] = IdComunidad.ToString();
        aux[1] = Nombre;
        aux[2] = Descripcion;
        aux[3] = NomTipo.Nombre;
        aux[4] = Lugar.Nombre;
        aux[5] = Lugar.Direccion;
        aux[6] = DiaReunion;
        return aux;
    }
}

```

- Entidad Confirmación

```

namespace Entidades
{
    public class Confirmacion
    {
        public int IdConfirmacion { get; set; }
        public Feligres NomFeligres { get; set; }
        public Templo NomTemplo { get; set; }
        public Templo TemploBau { get; set; }
        public int LibroBau { get; set; }
        public int FolioBau { get; set; }
        public DateTime FechaBau { get; set; }
        public DateTime Fecha { get; set; }
        public string Ministro { get; set; }
        public string PadrinoMadrina { get; set; }
        public LibroSacramento Libro { get; set; }
        public int Folio { get; set; }
        public int Numero { get; set; }
        public string Anotaciones { get; set; }

        public Confirmacion() { }

        public string[] DatosToArray()
        {
            string[] aux = new string[21];
            aux[0] = IdConfirmacion.ToString();
            aux[1] = NomFeligres.IdFeligres.ToString();
            aux[2] = NomFeligres.NomPersona.ApePaterno + " "
                    + NomFeligres.NomPersona.ApeMaterno + ", " +
                    NomFeligres.NomPersona.Nombres;
            aux[3] =
                NomFeligres.NomPersona.FecNac.ToString("dd/MM/yyyy");
            aux[4] = NomFeligres.Padre;
        }
    }
}

```

```

    aux[5] = NomFeligrés.Madre;
    aux[6] = NomTemplo.IdTemplo.ToString();
    aux[7] = NomTemplo.Nombre+ " de "
              "+NomTemplo.LugarPro.Nombre+
              "+NomTemplo.Distrito;
    aux[8] = TemploBau.IdTemplo.ToString();
    aux[9] = LibroBau.ToString();
    aux[10] = FolioBau.ToString();
    aux[11] = FechaBau.ToString("dd/MM/yyyy");
    aux[12] = Fecha.ToString("dd/MM/yyyy");
    aux[13] = Ministro;
    aux[14] = PadrinoMadrina;
    aux[15] = Libro.IdLibro.ToString();
    aux[16] = Folio.ToString();
    aux[17] = Numero.ToString();
    aux[18] = "Libro "+Libro.Numero+", folio "
              "+Folio+", número "+Numero;
    aux[19] = Anotaciones;
    aux[20] = Libro.Abrierto.ToString();
    return aux;
}
}

```

- Entidad Departamento

```
namespace Entidades
{
    public class Departamento
    {
        public int IdDepartamento { get; set; }
        public string Nombre { get; set; }
        public Pais NomPais { get; set; }

        public Departamento() { }

        public Departamento(int IdDepartamento) {
            this.IdDepartamento = IdDepartamento; }

        public Departamento(string Nombre, Pais NomPais) {
            this.Nombre = Nombre; this.NomPais = NomPais; }

    }
}
```

- Entidad Feligrés

```
namespace Entidades
{
    public class Feligres
    {
        public int IdFeligres { get; set; }
        public Persona NomPersona { get; set; }
        public Departamento LugNacDepa { get; set; }
        public Provincia LugNacPro { get; set; }
        public string DistritoFel { get; set; }
        public string Padre { get; set; }
        public Departamento NatPadreDepa { get; set; }
        public Provincia NatPadrePro { get; set; }
        public string DistritoPadre { get; set; }
        public string Madre { get; set; }
        public Departamento NatMadreDepa { get; set; }
        public Provincia NatMadrePro { get; set; }
    }
}
```

```

        public string DistritoMadre { get; set; }

        public Feligres() { }

        public string[] DatosToArray() {
            string[] aux = new string[22];
            aux[0]=IdFeligres.ToString();
            aux[1]=NomPersona.IdPersona.ToString();
            aux[2]=NomPersona.ApePaterno;
            aux[3]=NomPersona.ApeMaterno;
            aux[4]=NomPersona.Nombres;
            aux[5]=NomPersona.FecNac.ToString("dd/MM/yyyy");
            aux[6]=NomPersona.Sexo;
            aux[7]=LugNacDepa.NomPais.IdPais.ToString();
            aux[8]=LugNacPro.Nombre;
            aux[9]=LugNacDepa.Nombre;
            aux[10] = DistritoFel;
            string aux1 = (DistritoFel.Length>0) ? " - "
            "+DistritoFel : "" ;
            aux[11] = LugNacDepa.Nombre + ", " +
            LugNacPro.Nombre + aux1;
            aux[12]=Padre;
            aux[13]=NatPadreDepa.NomPais.IdPais.ToString();
            aux[14]=NatPadrePro.Nombre;
            aux[15] = NatPadreDepa.Nombre;
            aux[16] = DistritoPadre;
            aux[17] = Madre;
            aux[18] = NatMadreDepa.NomPais.IdPais.ToString();
            aux[19] = NatMadrePro.Nombre;
            aux[20] = NatMadreDepa.Nombre;
            aux[21] = DistritoMadre;
            return aux;
        }
    }
}

```

- Entidad LibroSacramento

```

namespace Entidades
{
    public class LibroSacramento
    {
        public int IdLibro { get; set; }
        public string Tipo { get; set; }
        public int Numero { get; set; }
        public int MaximoFolios { get; set; }
        public int NumeroPorFolio { get; set; }
        public int NumRegistros { get; set; }
        public DateTime FechaApertura { get; set; }
        public string FechaCierre { get; set; }
        public bool Abierto { get; set; }

        public LibroSacramento() { }

        public LibroSacramento(int IdLibro) { this.IdLibro =
        IdLibro; }

        public LibroSacramento(string Tipo) { this.Tipo =
        Tipo; }

        public LibroSacramento(int IdLibro, bool Abierto) {
        this.IdLibro = IdLibro; this.Abierto = Abierto; }
    }
}

```

```

        public string[] DatosToArray() {
            string[] aux = new string[9];
            aux[0] = IdLibro.ToString();
            aux[1] = Tipo;
            aux[2] = Numero.ToString();
            aux[3] = MaximoFolios.ToString();
            aux[4] = NumeroPorFolio.ToString();
            aux[5] = NumRegistros.ToString();
            aux[6] = FechaApertura.ToString("dd/MM/yyyy");
            aux[7] = (FechaCierre == null) ? "" :
                FechaCierre;
            aux[8] = (Abierto) ? "Abierto" : "Cerrado";
            return aux;
        }
    }
}

```

- Entidad Lugar

```

namespace Entidades
{
    public class Lugar
    {
        public int IdLugar { get; set; }
        public string Nombre { get; set; }
        public string Direccion { get; set; }

        public Lugar() { }

        public Lugar(int IdLugar) { this.IdLugar = IdLugar; }

        public string[] DatosToArray()
        {
            string[] aux = new string[3];
            aux[0] = IdLugar.ToString();
            aux[1] = Nombre;
            aux[2] = Direccion;
            return aux;
        }
    }
}

```

- Entidad Matrimonio

```

namespace Entidades
{
    public class Matrimonio
    {
        public int IdMatrimonio { get; set; }
        public Feligres Esposo { get; set; }
        public string EstCivilEsposo { get; set; }
        public int EdadEsposo { get; set; }
        public Templo TemBauEsposo { get; set; }
        public DateTime FecBauEsposo { get; set; }
        public Feligres Esposa { get; set; }
        public string EstCivilEsposa { get; set; }
        public int EdadEsposa { get; set; }
        public Templo TemBauEsposa { get; set; }
        public DateTime FecBauEsposa { get; set; }
        public Templo NomTemplo { get; set; }
        public DateTime Fecha { get; set; }
        public string Ministro { get; set; }
        public string Padrino { get; set; }
    }
}

```

```

        public string Madrina { get; set; }
        public string Testigo1 { get; set; }
        public string Testigo2 { get; set; }
        public LibroSacramento Libro { get; set; }
        public int Folio { get; set; }
        public int Numero { get; set; }
        public string Anotaciones { get; set; }

        public Matrimonio() { }

        public string[] DatosToArray()
        {
            string[] aux = new string[33];
            aux[0] = IdMatrimonio.ToString();
            aux[1] = Esposo.IdFeligres.ToString();
            aux[2] = Esposo.NomPersona.ApePaterno+
"+Esposo.NomPersona.ApeMaterno+", "+"
            Esposo.NomPersona.Nombres;
            aux[3] = Esposo.Padre;
            aux[4] = Esposo.Madre;
            aux[5] = EstCivilEsposo;
            aux[6] =
            Esposo.NomPersona.FecNac.ToString("dd/MM/yyyy");
            aux[7] = EdadEsposo.ToString();
            aux[8] = TemBauEsposo.IdTemplo.ToString();
            aux[9] = FecBauEsposo.ToString("dd/MM/yyyy");
            aux[10] = Esposa.IdFeligres.ToString();
            aux[11] = Esposa.NomPersona.ApePaterno+
"+Esposa.NomPersona.ApeMaterno+", "+"
            Esposa.NomPersona.Nombres;
            aux[12] = Esposa.Padre;
            aux[13] = Esposa.Madre;
            aux[14] = EstCivilEsposa;
            aux[15] =
            Esposa.NomPersona.FecNac.ToString("dd/MM/yyyy");
            aux[16] = EdadEsposa.ToString();
            aux[17] = TemBauEsposa.IdTemplo.ToString();
            aux[18] = FecBauEsposa.ToString("dd/MM/yyyy");
            aux[19] = NomTemplo.IdTemplo.ToString();
            aux[20] = NomTemplo.Nombre+ de
            "+NomTemplo.LugarPro.Nombre+ "+NomTemplo.Distrito;
            aux[21] = Fecha.ToString("dd/MM/yyyy");
            aux[22] = Ministro;
            aux[23] = Padrino;
            aux[24] = Madrina;
            aux[25] = Testigo1;
            aux[26] = Testigo2;
            aux[27] = Libro.IdLibro.ToString();
            aux[28] = Folio.ToString();
            aux[29] = Numero.ToString();
            aux[30] = "Libro "+Libro.Numero+, folio
            "+Folio+", numero "+Numero;
            aux[31] = Anotaciones;
            aux[32] = Libro.Abrierto.ToString();
            return aux;
        }
    }
}

```

- Entidad País  

```
namespace Entidades
{
```

```

public class Pais
{
    public int IdPais { get; set; }
    public string Nombre { get; set; }

    public Pais() { }

    public Pais(int IdPais) { this.IdPais = IdPais; }

    public Pais(int IdPais, string Nombre) { this.IdPais
= IdPais; this.Nombre = Nombre; }

}
}

```

- Entidad Persona

```

namespace Entidades
{
    public class Persona
    {
        public int IdPersona { get; set; }
        public string ApePaterno { get; set; }
        public string ApeMaterno { get; set; }
        public string Nombres { get; set; }
        public DateTime FecNac { get; set; }
        public string Sexo { get; set; }

        public Persona() { }

        public Persona(int IdPersona) { this.IdPersona =
IdPersona; }

        public Persona(int IdPersona, string ApePaterno,
string ApeMaterno, string Nombres,
DateTime FecNac, string Sexo)
        {
            this.IdPersona = IdPersona; this.ApePaterno =
ApePaterno; this.ApeMaterno = ApeMaterno;
            this.Nombres = Nombres; this.FecNac = FecNac;
            this.Sexo = Sexo;
        }
    }
}

```

- Entidad Personal

```

namespace Entidades
{
    public class Personal
    {
        public int IdPersonal { get; set; }
        public Persona NomPersona { get; set; }
        public Tipos NomTipo { get; set; }
        public string Domicilio { get; set; }
        public string NumTelefono { get; set; }
        public string NumCelular { get; set; }

        public Personal() { }

        public string[] DatosToArray()
        {

```

```

        string[] aux = new string[12];
        aux[0] = IdPersonal.ToString();
        aux[1] = NomPersona.IdPersona.ToString();
        aux[2] = NomPersona.ApePaterno;
        aux[3] = NomPersona.ApeMaterno;
        aux[4] = NomPersona.Nombres;
        aux[5] =
        NomPersona.FecNac.ToString("dd/MM/yyyy");
        aux[6] = NomPersona.Sexo;
        aux[7] = NomTipo.IdTipo.ToString();
        aux[8] = NomTipo.Nombre;
        aux[9] = Domicilio;
        aux[10] = NumTelefono;
        aux[11] = NumCelular;
        return aux;
    }

    public string[] DatosToArray2()
    {
        string[] aux = new string[6];
        aux[0] = NomPersona.ApePaterno + " " +
        NomPersona.ApeMaterno + ", " + NomPersona.Nombres;
        aux[1] =
        NomPersona.FecNac.ToString("dd/MM/yyyy");
        aux[2] = NomTipo.Nombre;
        aux[3] = Domicilio;
        aux[4] = NumTelefono;
        aux[5] = NumCelular;
        return aux;
    }
}
}

```

- Entidad Provincia

```

namespace Entidades
{
    public class Provincia
    {
        public int IdProvincia { get; set; }
        public string Nombre { get; set; }
        public Departamento NomDepartamento { get; set; }

        public Provincia() { }

        public Provincia(int IdProvincia) { this.IdProvincia
            = IdProvincia; }

        public Provincia(string Nombre) { this.Nombre =
            Nombre; }

    }
}

```

- Entidad Templo

```

namespace Entidades
{
    public class Templo
    {
        public int IdTemplo { get; set; }
        public string Nombre { get; set; }
        public string Tipo { get; set; }

```

```

        public Departamento LugarDepa { get; set; }
        public Provincia LugarPro { get; set; }
        public string Distrito { get; set; }

        public Templo() { }

        public Templo(string Nombre) { this.Nombre = Nombre;
    }

        public Templo(int IdTemplo) { this.IdTemplo =
IdTemplo; }

        public string[] DatosToArray() {
            string[] aux = new string[10];
            string r = (Distrito.Length == 0) ? "" : Distrito
+ " - ";
            aux[0] = IdTemplo.ToString();
            aux[1] = Nombre;
            aux[2] = Tipo;
            aux[3] = LugarDepa.NomPais.IdPais.ToString();
            aux[4] = LugarDepa.Nombre;
            aux[5] = LugarPro.Nombre;
            aux[6] = Distrito;
            aux[7] = r + LugarPro.Nombre + ", " +
LugarDepa.Nombre;
            return aux;
        }
    }
}

```

- Entidad Tipos

```

namespace Entidades
{
    public class Tipos
    {
        public int IdTipo { get; set; }
        public string Nombre { get; set; }
        public string Tipo { get; set; }

        public Tipos() { }

        public Tipos(int IdTipo, string Nombre, string Tipo)
        {
            this.IdTipo = IdTipo; this.Nombre = Nombre;
this.Tipo = Tipo;
        }

        public Tipos(string Tipo) { this.Tipo = Tipo; }

        public Tipos(int IdTipo) { this.IdTipo = IdTipo; }
    }
}

```

- Entidad Usuario

```

namespace Entidades
{
    public class Usuario
    {
        public int idusuario { get; set; }
        public string nick { get; set; }

```

```

        public string contraseña { get; set; }

        public Usuario() { }

    }
}

```

### Código fuente de la capa AccesoDatosNegocio

- ADNAsignación

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNAsignacion
    {
        static List<Personal> Secuencia(Asignacion e)
        {
            List<Personal> lista = new List<Personal>();
            using (MySqlConnection cn = new
            MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_SecuenciaAsginacion";
                    MySqlCommand cmd = new MySqlCommand(sql,
                    cn);
                    cmd.CommandType =
                    CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("id",
                    e.IdComunidad);
                    cn.Open();
                    MySqlDataReader dr = cmd.ExecuteReader();
                    while (dr.Read())
                    {
                        Personal p = new Personal();
                        p.IdPersonal =
                        Convert.ToInt32(dr[0]);
                        p.NomPersona = new Persona();
                        p.NomPersona.ApePaterno =
                        Convert.ToString(dr[1]);
                        p.NomPersona.ApeMaterno =
                        Convert.ToString(dr[2]);
                        p.NomPersona.Nombres =
                        Convert.ToString(dr[3]);
                        p.NomPersona.FecNac =
                        Convert.ToDateTime(dr[4]);
                        p.NomTipo = new Tipos();
                        p.NomTipo.Nombre =
                        Convert.ToString(dr[5]);
                        lista.Add(p);
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
                    null;
                }
            }
        }
    }
}

```

```

        }
        return lista;
    }

    static List<Personal> Secuencia2(Asignacion e)
    {
        List<Personal> lista = new List<Personal>();
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_SecuenciaAsginacion2";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
e.IdComunidad);
                cn.Open();
                MySqlDataReader dr = cmd.ExecuteReader();
                while (dr.Read())
                {
                    Personal p = new Personal();
                    p.NomPersona = new Persona();
                    p.NomPersona.ApePaterno =
Convert.ToString(dr[0]);
                    p.NomPersona.ApeMaterno =
Convert.ToString(dr[1]);
                    p.NomPersona.Nombres =
Convert.ToString(dr[2]);
                    p.NomPersona.FecNac =
Convert.ToDateTime(dr[3]);
                    p.NomTipo = new Tipos();
                    p.NomTipo.Nombre =
Convert.ToString(dr[4]);
                    p.Domicilio =
Convert.ToString(dr[5]);
                    p.NumTelefono =
Convert.ToString(dr[6]);
                    p.NumCelular =
Convert.ToString(dr[7]);
                    lista.Add(p);
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
null;
            }
        }
        return lista;
    }

    static bool AgregarAll(List<Asignacion> lista)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_InsertarAsginacion";

```

```

        MySqlCommand cmd ;
        cn.Open(); bool r = true;
        int idc = lista[0].IdComunidad;
        foreach (Asignacion fila in lista)
        {
            cmd = new MySqlCommand(sql, cn);
            cmd.CommandType =
                CommandType.StoredProcedure;

            cmd.Parameters.AddWithValue("idc", idc);

            cmd.Parameters.AddWithValue("idp", fila.IdPersonal);
            r = r &&
            Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        return r;
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
        return false;
    }
}
}

static bool Lleno(Asignacion e)
{
    using (MySqlConnection cn = new
    MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_AsignacionLlena";
            MySqlCommand cmd = new MySqlCommand(sql,
            cn);
            cmd.CommandType =
                CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idc",
            e.IdComunidad);
            cn.Open();
            return
            Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
            false;
        }
    }
}

static bool BorrarAll(Asignacion e)
{
    using (MySqlConnection cn = new
    MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeleteAsignacion";
            MySqlCommand cmd = new MySqlCommand(sql,
            cn);

```

```

        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idc",
e.IdComunidad);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
//Metodos Publicos
public static List<Personal> Lista(Asignacion e)
{
    return Secuencia(e);
}

public static List<Personal> Lista2(Asignacion e)
{
    return Secuencia2(e);
}

public static bool Guardar(List<Asignacion> lista)
{
    bool r = (Lleno(lista[0])) ? BorrarAll(lista[0])
: true;
    return (lista[0].IdPersonal == 0) ? r : r &&
AgregarAll(lista);
}

}
}

```

- ADNAudi

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNAudi
    {
        static bool Nuevo(Audi e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_InsertarAudi";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;

```

```

        cmd.Parameters.AddWithValue("des",
e.descripcion);
        cmd.Parameters.AddWithValue("fec",
DateTime.Now);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}

static bool Contar()
{
    using (MySqlConnection cn = new
MySqlConnection("Server=localhost;Database=dbParroquia;Uid=a
uditor;Pwd=audisistemas"))
    {
        try
        {
            string sql = "select ifnull(Count(*),0)
from information_schema.TABLES where TABLE_SCHEMA =
'dbparroquia';";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
//cmd.CommandType =
CommandType.StoredProcedure;
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

//Metodos publicos
public static bool Insertar(Audi e)
{
    return Nuevo(e);
}

public static bool ContarTablas()
{
    return Contar();
}
}

```

- ADNBautismo

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

```

```

namespace AccesoDatosNegocio
{
    public class ADNBautismo
    {
        static Bautismo CrearEntidad(MySqlDataReader lector)
        {
            Bautismo e = new Bautismo();
            e.IdBautismo = Convert.ToInt32(lector[0]);
            e.NomFeligres = new Feligres();
            e.NomFeligres.IdFeligres =
            Convert.ToInt32(lector[1]);
            e.NomFeligres.NomPersona = new Persona();
            e.NomFeligres.NomPersona.ApePaterno =
            Convert.ToString(lector[2]);
            e.NomFeligres.NomPersona.ApeMaterno =
            Convert.ToString(lector[3]);
            e.NomFeligres.NomPersona.Nombres =
            Convert.ToString(lector[4]);
            e.NomTemplo = new Templo();
            e.NomTemplo.IdTemplo =
            Convert.ToInt32(lector[5]);
            e.NomTemplo.Nombre = Convert.ToString(lector[6]);
            e.NomTemplo.LugarPro = new
            Provincia(Convert.ToString(lector[7]));
            e.NomTemplo.Distrito =
            Convert.ToString(lector[8]);
            e.Fecha = Convert.ToDateTime(lector[9]);
            e.Ministro = Convert.ToString(lector[10]);
            e.Padrino = Convert.ToString(lector[11]);
            e.Madrina = Convert.ToString(lector[12]);
            e.Libro = new LibroSacramento();
            e.Libro.IdLibro = Convert.ToInt32(lector[13]);
            e.Libro.Numero = Convert.ToInt32(lector[14]);
            e.Folio = Convert.ToInt32(lector[15]);
            e.Numero = Convert.ToInt32(lector[16]);
            e.NotaMarginal = Convert.ToString(lector[17]);
            e.Libro.Abrierto = Convert.ToBoolean(lector[18]);
            e.NomFeligres.NomPersona.Sexo =
            Convert.ToString(lector[19]);
            return e;
        }

        static List<Bautismo> Secuencia(Persona e)
        {
            List<Bautismo> lista = new List<Bautismo>();
            using (MySqlConnection cn = new
            MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_SecuenciaBautismo";
                    MySqlCommand cmd = new MySqlCommand(sql,
                    cn);
                    cmd.CommandType =
                    CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("apep",
                    e.ApePaterno);
                    cmd.Parameters.AddWithValue("apem",
                    e.ApeMaterno);
                    cn.Open();
                    MySqlDataReader dr = cmd.ExecuteReader();

```

```

                while (dr.Read())
                {
                    lista.Add(CrearEntidad(dr));
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
null;
            }
        }
        return lista;
    }

    static Bautismo CrearEntidad2(MySqlDataReader lector)
{
    Bautismo e = new Bautismo();
    e.Folio = Convert.ToInt32(lector["Folio"]);
    e.Numero = Convert.ToInt32(lector["Numero"]);
    return e;
}

    static Bautismo Secuencia2(LibroSacramento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_SecuenciaBautismo2";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdLibro);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            if (dr.Read())
            {
                return CrearEntidad2(dr);
            }
            else
            {
                return null;
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
null;
        }
    }
}

    static Bautismo CrearEntidad3(MySqlDataReader
lector,int op)
{
    Bautismo e = new Bautismo();
    e.NomTemplo = new Templo();
    e.NomTemplo.IdTemplo =
Convert.ToInt32(lector[0]);
}

```

```

e.NomTemplo.Nombre = Convert.ToString(lector[1]);
e.NomTemplo.Distrito =
Convert.ToString(lector[2]);
e.Fecha = Convert.ToDateTime(lector[3]);
e.Libro = new LibroSacramento();
e.Libro.Numero = Convert.ToInt32(lector[4]);
e.Folio = Convert.ToInt32(lector[5]);
if (op==1)
{
    e.Numero = Convert.ToInt32(lector[6]);
    e.Ministro = Convert.ToString(lector[7]);
    e.Padrino = Convert.ToString(lector[8]);
    e.Madrina = Convert.ToString(lector[9]);
    e.NotaMarginal =
Convert.ToString(lector[10]);
}
return e;
}

static Bautismo Secuencia3(Feligres e,int op)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = (op == 1) ?
"pa_SecuenciaBautismo31" : "pa_SecuenciaBautismo32";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idf",
e.IdFeligres);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            if (dr.Read())
            {
                return CrearEntidad3(dr, op);
            }
            else
            {
                return null;
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
null;
        }
    }
}

static bool Nuevo(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_InsertarBautismo";

```

```

        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idf",
e.NomFeligres.IdFeligres);
        cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
        cmd.Parameters.AddWithValue("fec",
e.Fecha);
        cmd.Parameters.AddWithValue("min",
e.Ministro);
        cmd.Parameters.AddWithValue("pad",
e.Padrino);
        cmd.Parameters.AddWithValue("mad",
e.Madrina);
        cmd.Parameters.AddWithValue("idl",
e.Libro.IdLibro);
        cmd.Parameters.AddWithValue("fol",
e.Folio);
        cmd.Parameters.AddWithValue("num",
e.Numero);
        cmd.Parameters.AddWithValue("nota",
e.NotaMarginal);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Actualizar(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdateBautismo";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
            cmd.Parameters.AddWithValue("fec",
e.Fecha);
            cmd.Parameters.AddWithValue("min",
e.Ministro);
            cmd.Parameters.AddWithValue("pad",
e.Padrino);
            cmd.Parameters.AddWithValue("mad",
e.Madrina);
            cmd.Parameters.AddWithValue("nota",
e.NotaMarginal);
            cmd.Parameters.AddWithValue("id",
e.IdBautismo);
        }
    }
}

```

```

                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool Existe(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteBautismo";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdBautismo);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool ExisteRegSac(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteRegSacBautismo";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idl",
e.Libro.IdLibro);
            cmd.Parameters.AddWithValue("fol",
e.Folio);
            cmd.Parameters.AddWithValue("num",
e.Numero);
            cmd.Parameters.AddWithValue("id",
e.IdBautismo);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
    }
}

```

```

        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }

    static string Fel(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_FelBautismo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idl",
e.Libro.IdLibro);
        cmd.Parameters.AddWithValue("fol",
e.Folio);
        cmd.Parameters.AddWithValue("num",
e.Numero);
        cn.Open(); MySqlDataReader dr =
cmd.ExecuteReader(); dr.Read();
        return Convert.ToString(dr[0]) + " " +
Convert.ToString(dr[1]) + ", " + Convert.ToString(dr[2]);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
"";
    }
}
}

static void Confi(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_ConfiBautismo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idf",
e.NomFeligres.IdFeligres);
        cn.Open();
        int idc =
Convert.ToInt32(cmd.ExecuteScalar());
        if (idc > 0)
        {
            sql =
"pa_UpdateConfirmacionBautismo";
            cmd = new MySqlCommand(sql, cn);
        }
    }
}
}

```

```

        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
        cmd.Parameters.AddWithValue("fec",
e.Fecha);
        cmd.Parameters.AddWithValue("id",
idc);
        cmd.ExecuteNonQuery();
    }
}
catch (Exception ex)
{
    MessageBox.Show(ex.ToString());
}
}

static void Matri(Bautismo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql =
(e.NomFeligres.NomPersona.Sexo.Equals("Masculino")) ?
"pa_MatriBautismo1" :
"pa_MatriBautismo2";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idf",
e.NomFeligres.IdFeligres);
            cn.Open();
            int idm =
Convert.ToInt32(cmd.ExecuteScalar());
            if (idm > 0)
            {
                sql =
(e.NomFeligres.NomPersona.Sexo.Equals("Masculino")) ?
"pa_UpdateMatrimonioBautismo1" :
"pa_UpdateMatrimonioBautismo2";
                cmd = new MySqlCommand(sql, cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
                cmd.Parameters.AddWithValue("fec",
e.Fecha);
                cmd.Parameters.AddWithValue("id",
idm);
                cmd.ExecuteNonQuery();
            }
        }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
}

```

```

//Metodos Publicos
public static List<Bautismo> Lista(Persona e)
{
    return Secuencia(e);
}

public static Bautismo FolioNumero(LibroSacramento
e)
{
    return Secuencia2(e);
}

public static bool Guardar(Bautismo e)
{
    if (Existe(e))
    {
        if (Actualizar(e))
        {
            Confi(e); Matri(e); return true;
        }
        else
        {
            return false;
        }
    }
    else
    {
        if (Nuevo(e))
        {
            ADNLibroSacramento.Comprobar(e.Libro);
            return true;
        }
        else
        {
            return false;
        }
    }
}

public static bool ExisteLFN(Bautismo e)
{
    return ExisteRegSac(e);
}

public static string FeligresLFN(Bautismo e)
{
    return Fel(e);
}

public static Bautismo Datos(Feligres e,int opcion)
{
    return Secuencia3(e,opcion);
}
}
}

```

- ADNComunidad

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;

```

```

using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNComunidad
    {
        static Comunidad CrearEntidad(MySqlDataReader
lector)
        {
            Comunidad e = new Comunidad();
            e.IdComunidad =
Convert.ToInt32(lector["IdComunidad"]);
            e.Nombre = Convert.ToString(lector["Nombre"]);
            e.Descripcion =
Convert.ToString(lector["Descripcion"]);
            e.NomTipo = new Tipos();
            e.NomTipo.Nombre = Convert.ToString(lector[3]);
            e.Lugar = new Lugar();
            e.Lugar.Nombre = Convert.ToString(lector[4]);
            e.Lugar.Direccion = Convert.ToString(lector[5]);
            e.DiaReunion =
Convert.ToString(lector["DiaReunion"]);
            return e;
        }

        static List<Comunidad> Secuencia(Comunidad e)
        {
            List<Comunidad> lista = new List<Comunidad>();
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_SecuenciaComunidad";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;

                    cmd.Parameters.AddWithValue("nom", e.Nombre);
                    cn.Open();
                    MySqlDataReader dr =
cmd.ExecuteReader();
                    while (dr.Read())
                    {
                        lista.Add(CrearEntidad(dr));
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
null;
                }
            }
            return lista;
        }

        static List<Comunidad> Secuencia2(Comunidad e)
        {
            List<Comunidad> lista = new List<Comunidad>();
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))

```

```

        {
            try
            {
                string sql = "pa_SecuenciaComunidad2";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("nom",
                e.Nombre);
                cn.Open();
                MySqlDataReader dr =
                cmd.ExecuteReader();
                while (dr.Read())
                {
                    Comunidad c = new Comunidad();
                    c.IdComunidad =
                    Convert.ToInt32(dr[0]);
                    c.Nombre = Convert.ToString(dr[1]);
                    lista.Add(c);
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                null;
            }
        }
        return lista;
    }

    static bool Nuevo(Comunidad e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_InsertarComunidad";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("idt",
                e.NomTipo.IdTipo);
                cmd.Parameters.AddWithValue("idl",
                e.Lugar.IdLugar);
                cmd.Parameters.AddWithValue("nom",
                e.Nombre);
                cmd.Parameters.AddWithValue("des",
                e.Descripcion);
                cmd.Parameters.AddWithValue("dia",
                e.DiaReunion);
                cn.Open();
                return
                Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                false;
            }
        }
    }
}

```

```

        }
    }

    static bool Actualizar(Comunidad e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_UpdateComunidad";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("idt",
                e.NomTipo.IdTipo);
                cmd.Parameters.AddWithValue("idl",
                e.Lugar.IdLugar);
                cmd.Parameters.AddWithValue("nom",
                e.Nombre);
                cmd.Parameters.AddWithValue("des",
                e.Descripcion);
                cmd.Parameters.AddWithValue("dia",
                e.DiaReunion);
                cmd.Parameters.AddWithValue("id",
                e.IdComunidad);
                cn.Open();
                return
                Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                false;
            }
        }
    }

    static bool Existe(Comunidad e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteComunidad";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
                e.IdComunidad);
                cn.Open();
                return
                Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                false;
            }
        }
    }
}

```

```

        }
    }

    static bool ExisteNom(Comunidad e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteNomComunidad";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("nom",
e.Nombre);
                cmd.Parameters.AddWithValue("id",
e.IdComunidad);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool Borrar(Comunidad e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_DeleteComunidad";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
e.IdComunidad);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static string tipo(Comunidad e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))

```

```

        {
            try
            {
                string sql = "pa_TipoComunidad";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
                e.IdComunidad);
                cn.Open();
                MySqlDataReader dr =
                cmd.ExecuteReader();
                if (dr.Read())
                {
                    return Convert.ToString(dr[0]);
                }
                else
                {
                    return "";
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                null;
            }
        }
    }

    //Metodos publicos
    public static List<Comunidad> Lista(Comunidad e)
    {
        return Secuencia(e);
    }

    public static List<Comunidad> Lista2(Comunidad e)
    {
        return Secuencia2(e);
    }

    public static bool Guardar(Comunidad e)
    {
        if (Existe(e))
        {
            return Actualizar(e);
        }
        else
        {
            return Nuevo(e);
        }
    }

    public static bool ExisteNombre(Comunidad e)
    {
        return ExisteNom(e);
    }

    public static bool Eliminar(Comunidad e)
    {
        return Borrar(e);
    }
}

```

```

        public static string Tipo(Comunidad e)
        {
            return tipo(e);
        }
    }
}

• ADNConfirmación
using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNConfirmacion
    {
        static Confirmacion CrearEntidad(MySqlDataReader lector)
        {
            Confirmacion e = new Confirmacion();
            e.IdConfirmacion = Convert.ToInt32(lector[0]);
            e.NomFeligres = new Feligres();
            e.NomFeligres.IdFeligres =
            Convert.ToInt32(lector[1]);
            e.NomFeligres.NomPersona = new Persona();
            e.NomFeligres.NomPersona.ApePaterno =
            Convert.ToString(lector[2]);
            e.NomFeligres.NomPersona.ApeMaterno =
            Convert.ToString(lector[3]);
            e.NomFeligres.NomPersona.Nombres =
            Convert.ToString(lector[4]);
            e.NomFeligres.NomPersona.FecNac =
            Convert.ToDateTime(lector[5]);
            e.NomFeligres.Padre =
            Convert.ToString(lector[6]);
            e.NomFeligres.Madre =
            Convert.ToString(lector[7]);
            e.NomTemplo = new Templo();
            e.NomTemplo.IdTemplo =
            Convert.ToInt32(lector[8]);
            e.NomTemplo.Nombre = Convert.ToString(lector[9]);
            e.NomTemplo.LugarPro = new
            Provincia(Convert.ToString(lector[10]));
            e.NomTemplo.Distrito =
            Convert.ToString(lector[11]);
            e.TemploBau = new
            Templo(Convert.ToInt32(lector[12]));
            e.LibroBau = Convert.ToInt32(lector[13]);
            e.FolioBau = Convert.ToInt32(lector[14]);
            e.FechaBau = Convert.ToDateTime(lector[15]);
            e.Fecha = Convert.ToDateTime(lector[16]);
            e.Ministro = Convert.ToString(lector[17]);
            e.PadrinoMadrina = Convert.ToString(lector[18]);
            e.Libro = new LibroSacramento();
            e.Libro.IdLibro = Convert.ToInt32(lector[19]);
            e.Libro.Numero = Convert.ToInt32(lector[20]);
            e.Folio = Convert.ToInt32(lector[21]);
            e.Numero = Convert.ToInt32(lector[22]);
            e.Anotaciones = Convert.ToString(lector[23]);
        }
    }
}

```

```

        e.Libro.Aberto = Convert.ToBoolean(lector[24]);
        return e;
    }

    static List<Confirmacion> Secuencia(Persona e)
    {
        List<Confirmacion> lista = new
List<Confirmacion>();
        using (MySqlConnection cn = new
MySqlConnection(Cexion.Cadena))
        {
            try
            {
                string sql = "pa_SecuenciaConfirmacion";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("apep",
e.ApePaterno);
                cmd.Parameters.AddWithValue("apem",
e.ApeMaterno);
                cn.Open();
                MySqlDataReader dr = cmd.ExecuteReader();
                while (dr.Read())
                {
                    lista.Add(CrearEntidad(dr));
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
null;
            }
        }
        return lista;
    }

    static Confirmacion CrearEntidad2(MySqlDataReader
lector)
{
    Confirmacion e = new Confirmacion();
    e.NomTemplo = new Templo();
    e.NomTemplo.Nombre = Convert.ToString(lector[0]);
    e.NomTemplo.Distrito =
Convert.ToString(lector[1]);
    e.TemploBau = new Templo();
    e.TemploBau.IdTemplo =
Convert.ToInt32(lector[2]);
    e.TemploBau.Nombre = Convert.ToString(lector[3]);
    if (Convert.ToInt32(lector[4]) == 27)
    {
        e.TemploBau.LugarDepa = new
Departamento(Convert.ToString(lector[6]),
            new Pais(Convert.ToInt32(lector[4])));
        e.TemploBau.LugarPro = new
Provincia(Provincia(e.TemploBau.IdTemplo));
    }
    else
    {
        e.TemploBau.LugarDepa = new
Departamento(Convert.ToString(lector[5]),

```

```

                new Pais(Convert.ToInt32(lector[4]));
                e.TemploBau.LugarPro = new
                Provincia(Convert.ToString(lector[6]));
            }

            e.Fecha = Convert.ToDateTime(lector[7]);
            e.Ministro = Convert.ToString(lector[8]);
            e.PadrinoMadrina = Convert.ToString(lector[9]);
            e.Anotaciones = Convert.ToString(lector[10]);
            e.Libro = new LibroSacramento();
            e.Libro.Numero = Convert.ToInt32(lector[11]);
            e.Folio = Convert.ToInt32(lector[12]);
            e.Numero = Convert.ToInt32(lector[13]);
            return e;
        }

        static Confirmacion Secuencia2(Feligres e)
        {
            using (MySqlConnection cn = new
            MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_SecuenciaConfirmacion2";
                    MySqlCommand cmd = new MySqlCommand(sql,
                    cn);
                    cmd.CommandType =
                    CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("idf",
                    e.IdFeligres);
                    cn.Open();
                    MySqlDataReader dr = cmd.ExecuteReader();
                    if (dr.Read())
                    {
                        return CrearEntidad2(dr);
                    }
                    else
                    {
                        return null;
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
                    null;
                }
            }
        }

        static string Provincia(int idtemplo)
        {
            using (MySqlConnection cn = new
            MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_ProvinciaTemplo";
                    MySqlCommand cmd = new MySqlCommand(sql,
                    cn);
                    cmd.CommandType =
                    CommandType.StoredProcedure;

```

```

        cmd.Parameters.AddWithValue("id",
idtemplo);
        cn.Open();
        MySqlDataReader dr = cmd.ExecuteReader();
        dr.Read();
        return Convert.ToString(dr[0]);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}

static bool Nuevo(Confirmacion e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_InsertarConfirmacion";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idf",
e.NomFeligres.IdFeligres);
            cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
            cmd.Parameters.AddWithValue("tb",
e.TemploBau.IdTemplo);
            cmd.Parameters.AddWithValue("libbau",
e.LibroBau);
            cmd.Parameters.AddWithValue("folbau",
e.FolioBau);
            cmd.Parameters.AddWithValue("fecbau",
e.FechaBau);
            cmd.Parameters.AddWithValue("fec",
e.Fecha);
            cmd.Parameters.AddWithValue("min",
e.Ministro);
            cmd.Parameters.AddWithValue("padmad",
e.PadrinoMadrina);
            cmd.Parameters.AddWithValue("idl",
e.Libro.IdLibro);
            cmd.Parameters.AddWithValue("fol",
e.Folio);
            cmd.Parameters.AddWithValue("num",
e.Numero);
            cmd.Parameters.AddWithValue("anot",
e.Anotaciones);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

```

```

        }
    }

    static bool Actualizar(Confirmacion e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Cexion.Cadena))
        {
            try
            {
                string sql = (e.TemploBau.IdTemplo == 1)
                ? "pa_UpdateConfirmacion1" : "pa_UpdateConfirmacion2";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("idt",
                e.NomTemplo.IdTemplo);
                if (e.TemploBau.IdTemplo > 1)
                {
                    cmd.Parameters.AddWithValue("tb",
                    e.TemploBau.IdTemplo);
                    cmd.Parameters.AddWithValue("libbau",
                    e.LibroBau);
                    cmd.Parameters.AddWithValue("folbau",
                    e.FolioBau);
                    cmd.Parameters.AddWithValue("fecbau",
                    e.FechaBau);
                }
                cmd.Parameters.AddWithValue("fec",
                e.Fecha);
                cmd.Parameters.AddWithValue("min",
                e.Ministro);
                cmd.Parameters.AddWithValue("padmad",
                e.PadrinoMadrina);
                cmd.Parameters.AddWithValue("anot",
                e.Anotaciones);
                cmd.Parameters.AddWithValue("id",
                e.IdConfirmacion);
                cn.Open();
                return
                Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                false;
            }
        }
    }

    static bool Existe(Confirmacion e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Cexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteConfirmacion";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);

```

```

        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("id",
e.IdConfirmacion);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}

static bool ExisteRegSac(Confirmacion e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql =
"pa_ExisteRegSacConfirmacion";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idl",
e.Libro.IdLibro);
            cmd.Parameters.AddWithValue("fol",
e.Folio);
            cmd.Parameters.AddWithValue("num",
e.Numero);
            cmd.Parameters.AddWithValue("id",
e.IdConfirmacion);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static string Fel(Confirmacion e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_FelConfirmacion";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;

```



- ADNCrearBaseDatos

```

using MySql.Data.MySqlClient;
using MySql.Data.Types;
using System.Windows.Forms;

namespace AccesoDatosNegocio
{
    public class ADNCrearBaseDatos
    {
        static bool reiniciar()
        {
            using (MySqlConnection cn = new MySqlConnection("Server=localhost;Database=dbParroquia;Uid=root;Pwd=a1c9c9a0"))
            {
                try
                {
                    MySqlCommand cmd = new MySqlCommand("DROP DATABASE IF EXISTS dbParroquia;", cn);
                    cn.Open();
                    cmd.ExecuteNonQuery();
                    cmd = new MySqlCommand("CREATE DATABASE dbParroquia; use dbParroquia;", cn);
                    return Convert.ToBoolean(cmd.ExecuteNonQuery());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return false;
                }
            }
        }

        public static bool CrearBaseDatos()
        {
            return reiniciar();
        }
    }
}

```

- ADNDepartamento

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNDepartamento
    {
        static Departamento CrearEntidad(MySqlDataReader lector)
        {
            Departamento e = new Departamento();
            e.IdDepartamento =
Convert.ToInt32(lector["IdDepartamento"]);
            e.Nombre = Convert.ToString(lector["Nombre"]);
            return e;
        }
}

```

```

        static List<Departamento> Secuencia(Pais e)
        {
            List<Departamento> lista = new
List<Departamento>();
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_SecuenciaDepartamento";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("id",
e.IdPais);
                    cn.Open();
                    MySqlDataReader dr = cmd.ExecuteReader();
                    while (dr.Read())
                    {
                        lista.Add(CrearEntidad(dr));
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
null;
                }
            }
            return lista;
        }

        static bool Nuevo(Departamento e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_InsertarDepartamento";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    cmd.Parameters.AddWithValue("idp",
e.NomPais.IdPais);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteNonQuery());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool Actualizar(Departamento e)
    
```

```

        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_UpdateDepartamento";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    cmd.Parameters.AddWithValue("id",
e.IdDepartamento);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteNonQuery());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool Existe(Departamento e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_ExisteDepartamento";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("id",
e.IdDepartamento);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteScalar());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool ExisteNom(Departamento e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_ExisteNomDepartamento";

```

```

        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("nom",
e.Nombre);
        cmd.Parameters.AddWithValue("idd",
e.IdDepartamento);
        cmd.Parameters.AddWithValue("idp",
e.NomPais.IdPais);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Borrar(Departamento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeleteDepartamento";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdDepartamento);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

//Metodos publicos
public static List<Departamento> Lista(Pais e)
{
    return Secuencia(e);
}

public static bool Guardar(Departamento e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {

```

```

                return Nuevo(e);
            }
        }

        public static bool ExisteNombre(Departamento e)
        {
            return ExisteNom(e);
        }

        public static bool Eliminar(Departamento e)
        {
            return Borrar(e);
        }
    }
}

```

- ADNFeligres

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNFeligres
    {
        static Feligres CrearEntidad(MySqlDataReader lector)
        {
            Feligres e = new Feligres();
            e.IdFeligres =
Convert.ToInt32(lector["IdFeligres"]);
            e.NomPersona = new
Persona(Convert.ToInt32(lector["IdPersona"]),

Convert.ToString(lector["ApePaterno"]),Convert.ToString(lecto
r["ApeMaterno"]),

Convert.ToString(lector["Nombres"]),Convert.ToDateTime(lecto
r["FecNac"]),
            Convert.ToString(lector["Sexo"]));
//lector[7]=pa1.IdPaís
            if (Convert.ToInt32(lector[7]) == 27)
            {
                e.LugNacDepa = new
Departamento(Convert.ToString(lector[9]),
                new País(Convert.ToInt32(lector[7])));
                e.LugNacPro = new
Provincia(Provincia(e.IdFeligres, 1));
            }
            else
            {
                e.LugNacDepa = new
Departamento(Convert.ToString(lector[8]),
                new País(Convert.ToInt32(lector[7])));
                e.LugNacPro = new
Provincia(Convert.ToString(lector[9]));
            }
            e.DistritoFel =
Convert.ToString(lector["DistritoFel"]);
            e.Padre = Convert.ToString(lector["Padre"]);
        }
    }
}

```

```

        //lector[12]=pa2.IdPais
        if (e.Padre=="-----")
        {
            e.NatPadreDepa = new Departamento("", new
Pais(0));
            e.NatPadrePro = new Provincia("");
            e.DistritoPadre = "";
        }
        else
        {
            Feligres pa = padre(e.IdFeligres);
            if (pa.NatPadreDepa.NomPais.IdPais == 27)
            {
                e.NatPadreDepa = new
Departamento(pa.NatPadreDepa.Nombre,
new
Pais(pa.NatPadreDepa.NomPais.IdPais));
                e.NatPadrePro = new
Provincia(Provincia(e.IdFeligres, 2));
            }
            else
            {
                e.NatPadreDepa = new
Departamento(pa.NatPadreDepa.NomPais.Nombre,
new
Pais(pa.NatPadreDepa.NomPais.IdPais));
                e.NatPadrePro = new
Provincia(pa.NatPadreDepa.Nombre);
            }
            e.DistritoPadre = pa.DistritoPadre;
        }

        e.Madre = Convert.ToString(lector["Madre"]);

        if (Convert.ToInt32(lector[13]) == 27)
        {
            e.NatMadreDepa = new
Departamento(Convert.ToString(lector[15]),
new Pais(Convert.ToInt32(lector[13])));
            e.NatMadrePro = new
Provincia(Provincia(e.IdFeligres, 3));
        }
        else
        {
            e.NatMadreDepa = new
Departamento(Convert.ToString(lector[14]),
new Pais(Convert.ToInt32(lector[13])));
            e.NatMadrePro = new
Provincia(Convert.ToString(lector[15]));
        }
        e.DistritoMadre =
Convert.ToString(lector["DistritoMadre"]);
        return e;
    }

    static string Provincia(int idfeligres,int op)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {

```

```

        try
        {
            string sql = (op == 1) ?
"pa_ProvinciaFeligres1" :
((op == 2) ?
"pa_ProvinciaFeligres2" : "pa_ProvinciaFeligres3");
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
idfeligres);
            cn.Open();
            MySqlDataReader dr =
cmd.ExecuteReader();
            dr.Read();
            return Convert.ToString(dr[0]);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
null;
        }
    }

    static Feligres padre(int idfeligres)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            Feligres f = new Feligres();
            try
            {
                string sql = "pa_PadreFeligres";
                MySqlCommand cmd = new
MySqlCommand(sql, cn);
                cmd.CommandType =
CommandType.StoredProcedure;

                cmd.Parameters.AddWithValue("id", idfeligres);
                cn.Open();
                MySqlDataReader dr =
cmd.ExecuteReader();
                dr.Read();
                f.NatPadreDepa = new Departamento();
                f.NatPadreDepa.Nombre =
Convert.ToString(dr[2]);
                f.NatPadreDepa.NomPais = new Pais();
                f.NatPadreDepa.NomPais.IdPais =
Convert.ToInt32(dr[0]);
                f.NatPadreDepa.NomPais.Nombre =
Convert.ToString(dr[1]);
                f.DistritoPadre =
Convert.ToString(dr[3]);
                return f;
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
null;
            }
        }
    }
}

```

```

        }
    }

    static List<Feligres> Secuencia(Persona e,int op)
    {
        List<Feligres> lista = new List<Feligres>();
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_SecuenciaFeligres";
                if (op == 1)
                {
                    sql = "pa_SecuenciaFeligres1";
                }
                else if (op == 2)
                {
                    sql = "pa_SecuenciaFeligres2";
                }
                else if (op == 3)
                {
                    sql = (e.Sexo == "Masculino") ?
"pa_SecuenciaFeligres3" : "pa_SecuenciaFeligres32";
                }
                else if (op==4)
                {
                    sql = "pa_SecuenciaFeligres4";
                }
                MySqlCommand cmd = new
                MySqlCommand(sql, cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("apep",
e.ApePaterno);
                cmd.Parameters.AddWithValue("apem",
e.ApeMaterno);
                cn.Open();
                MySqlDataReader dr =
                cmd.ExecuteReader();
                while (dr.Read())
                {
                    lista.Add(CrearEntidad(dr));
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                null;
            }
        }
        return lista;
    }

    static Feligres Datos(Feligres e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_DatosFeligres";

```

```

        MySqlCommand cmd = new
MySqlCommand(sql, cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idf",
e.IdFeligres);
        cn.Open(); MySqlDataReader dr =
cmd.ExecuteReader(); dr.Read();
        return CrearEntidad(dr);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
}

static bool Nuevo(Feligres e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_InsertarFeligres";
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idper",
e.NomPersona.IdPersona);
            cmd.Parameters.AddWithValue("lnd",
e.LugNacDepa.IdDepartamento);

            if (e.LugNacPro.IdProvincia > 0)
            {
                cmd.Parameters.AddWithValue("lnp",
e.LugNacPro.IdProvincia);
            }
            else
            {
                cmd.Parameters.AddWithValue("lnp",
DBNull.Value);
            }

            cmd.Parameters.AddWithValue("disfel",
e.DistritoFel);

            cmd.Parameters.AddWithValue("padre",
e.Padre);

            if (e.NatPadreDepa.IdDepartamento > 0)
            {
                cmd.Parameters.AddWithValue("npd",
e.NatPadreDepa.IdDepartamento);
            }
            else
            {
                cmd.Parameters.AddWithValue("npd",
DBNull.Value);
            }
        }
    }
}

```

```

        if (e.NatPadrePro.IdProvincia > 0)
        {
            cmd.Parameters.AddWithValue("npp",
e.NatPadrePro.IdProvincia);
        }
        else
        {
            cmd.Parameters.AddWithValue("npp",
DBNull.Value);
        }

        cmd.Parameters.AddWithValue("dispa",
e.DistritoPadre);
        cmd.Parameters.AddWithValue("madre",
e.Madre);
        cmd.Parameters.AddWithValue("nmd",
e.NatMadreDepa.IdDepartamento);

        if (e.NatMadrePro.IdProvincia > 0)
        {
            cmd.Parameters.AddWithValue("nmp",
e.NatMadrePro.IdProvincia);
        }
        else
        {
            cmd.Parameters.AddWithValue("nmp",
DBNull.Value);
        }

        cmd.Parameters.AddWithValue("disma",
e.DistritoMadre);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
        return false;
    }
}
}

static bool Actualizar(Feligres e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdateFeligres";
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("lnd",
e.LugNacDepa.IdDepartamento);

            if (e.LugNacPro.IdProvincia > 0)
            {
                cmd.Parameters.AddWithValue("lnp",
e.LugNacPro.IdProvincia);
            }
        }
    }
}

```

```

        }
        else
        {
            cmd.Parameters.AddWithValue("lnp",
DBNull.Value);
        }

        cmd.Parameters.AddWithValue("disfel",
e.DistritoFel);
        cmd.Parameters.AddWithValue("padre",
e.Padre);

        if (e.NatPadreDepa.IdDepartamento > 0)
        {
            cmd.Parameters.AddWithValue("npd",
e.NatPadreDepa.IdDepartamento);
        }
        else
        {
            cmd.Parameters.AddWithValue("npd",
DBNull.Value);
        }

        if (e.NatPadrePro.IdProvincia > 0)
        {
            cmd.Parameters.AddWithValue("npp",
e.NatPadrePro.IdProvincia);
        }
        else
        {
            cmd.Parameters.AddWithValue("npp",
DBNull.Value);
        }

        cmd.Parameters.AddWithValue("dispa",
e.DistritoPadre);
        cmd.Parameters.AddWithValue("madre",
e.Madre);
        cmd.Parameters.AddWithValue("nmd",
e.NatMadreDepa.IdDepartamento);

        if (e.NatMadrePro.IdProvincia > 0)
        {
            cmd.Parameters.AddWithValue("nmp",
e.NatMadrePro.IdProvincia);
        }
        else
        {
            cmd.Parameters.AddWithValue("nmp",
DBNull.Value);
        }

        cmd.Parameters.AddWithValue("disma",
e.DistritoMadre);
        cmd.Parameters.AddWithValue("id",
e.IdFeligres);

        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {

```

```

        MessageBox.Show(ex.ToString()); return
    false;
}
}
}

static bool ExisteAll(Feligres e, Persona p)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteAllFeligres";
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;

            cmd.Parameters.AddWithValue("apep", p.ApePaterno);
            cmd.Parameters.AddWithValue("apem",
p.ApeMaterno);
            cmd.Parameters.AddWithValue("nom",
p.Nombres);
            cmd.Parameters.AddWithValue("fec",
p.FecNac);
            cmd.Parameters.AddWithValue("lnd",
e.LugNacDepa.IdDepartamento);
            cmd.Parameters.AddWithValue("padre",
e.Padre);
            cmd.Parameters.AddWithValue("madre",
e.Madre);
            cmd.Parameters.AddWithValue("nmd",
e.NatMadreDepa.IdDepartamento);

            cmd.Parameters.AddWithValue("id", e.IdFeligres);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Existe(Feligres e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteFeligres";
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdFeligres);
        }
    }
}

```

```

        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}

static bool Borrar(Feligres e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeleteFeligres";
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdFeligres);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static Feligres LNac(Feligres e)
{
    Feligres f = new Feligres();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_LNacFeligres";
            MySqlCommand cmd = new
MySqlCommand(sql, cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdFeligres);
            cn.Open(); MySqlDataReader dr =
cmd.ExecuteReader(); dr.Read();

            if (Convert.ToInt32(dr[0]) == 27)
            {
                f.LugNacDepa = new
Departamento(Convert.ToString(dr[2]),
new
Pais(Convert.ToInt32(dr[0]))));
            }
        }
    }
}

```

```

                f.LugNacPro = new
Provincia(Provincia(e.IdFeligres, 1));
            }
        else
        {
            f.LugNacDepa = new
Departamento(Convert.ToString(dr[1]),
new
Pais(Convert.ToInt32(dr[0])));
            f.LugNacPro = new
Provincia(Convert.ToString(dr[2]));
        }
        f.DistritoFel =
Convert.ToString(dr["DistritoFel"]);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
return f;
}

//Metodos Publicos
public static List<Feligres> Lista(Persona e,int
opcion)
{
    return Secuencia(e,opcion);
}

public static bool Guardar(Feligres e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

public static bool Eliminar(Feligres e)
{
    if (Borrar(e))
    {
        return ADNPersona.Eliminar(new
Persona(e.NomPersona.IdPersona));
    }
    else
    {
        return false;
    }
}

public static Feligres DatosFel(Feligres e)
{
    return Datos(e);
}

public static Feligres LugarNac(Feligres e)

```

```

        {
            return LNac(e);
        }

        public static bool ExisteAllFeligres(Feligres
e, Persona p)
{
    return ExisteAll(e, p);
}
}

● ADNLibroSacramento
using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNLibroSacramento
    {
        static LibroSacramento CrearEntidad(MySqlDataReader
lector)
        {
            LibroSacramento e = new LibroSacramento();
            e.IdLibro = Convert.ToInt32(lector["IdLibro"]);
            e.Tipo = Convert.ToString(lector["Tipo"]);
            e.Numero = Convert.ToInt32(lector["Numero"]);
            e.MaximoFolios =
Convert.ToInt32(lector["MaximoFolios"]);
            e.NumeroPorFolio =
Convert.ToInt32(lector["NumeroPorFolio"]);
            e.NumRegistros =
Convert.ToInt32(lector["NumRegistros"]);
            e.FechaApertura =
Convert.ToDateTime(lector["FechaApertura"]);
            if (lector["FechaCierre"]==DBNull.Value)
            {
                e.FechaCierre = null;
            }
            else
            {
                e.FechaCierre =
Convert.ToDateTime(lector["FechaCierre"]).ToString("dd/MM/yyyy");
            }
            e.Abrierto = Convert.ToBoolean(lector["Abierto"]);
            return e;
        }

        static List<LibroSacramento>
Secuencia(LibroSacramento e)
{
    List<LibroSacramento> lista = new
List<LibroSacramento>();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {

```

```

        string sql = "pa_SecuenciaLibro";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("tip",
e.Tipo);
        cn.Open();
        MySqlDataReader dr = cmd.ExecuteReader();
        while (dr.Read())
        {
            lista.Add(CrearEntidad(dr));
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
return lista;
}

static LibroSacramento CrearEntidad2(MySqlDataReader
lector)
{
    LibroSacramento e = new LibroSacramento();
    e.IdLibro = Convert.ToInt32(lector["IdLibro"]);
    e.Numero = Convert.ToInt32(lector["Numero"]);
    return e;
}

static List<LibroSacramento>
Secuencia2(LibroSacramento e)
{
    List<LibroSacramento> lista = new
List<LibroSacramento>();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = (e.IdLibro > 0) ?
"pa_SecuenciaLibro21" : "pa_SecuenciaLibro22";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            if (e.IdLibro > 0)
            {
                cmd.Parameters.AddWithValue("id",
e.IdLibro);
            }
            else
            {
                cmd.Parameters.AddWithValue("tip",
e.Tipo);
                cmd.Parameters.AddWithValue("est",
true);
            }
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();

```

```

                while (dr.Read())
                {
                    lista.Add(CrearEntidad2(dr));
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
            null;
            }
        }
        return lista;
    }

    static LibroSacramento CrearEntidad3(MySqlDataReader
lector)
{
    LibroSacramento e = new LibroSacramento();
    e.MaximoFolios = Convert.ToInt32(lector[0]);
    e.NumeroPorFolio = Convert.ToInt32(lector[1]);
    e.NumRegistros = Convert.ToInt32(lector[2]);
    return e;
}

static LibroSacramento Secuencia3(LibroSacramento
e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_SecuenciaLibro3";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdLibro);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            dr.Read(); return CrearEntidad3(dr);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
            null;
        }
    }
}

static bool LibroOpen(LibroSacramento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_LibroOpen";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);

```

```

        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("tip",
e.Tipo);
        cmd.Parameters.AddWithValue("est", true);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Nuevo(LibroSacramento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_InsertarLibro";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("tip",
e.Tipo);
            cmd.Parameters.AddWithValue("num",
e.Numero);
            cmd.Parameters.AddWithValue("max",
e.MaximoFolios);
            cmd.Parameters.AddWithValue("numfolio",
e.NumeroPorFolio);
            cmd.Parameters.AddWithValue("reg", 0);
            cmd.Parameters.AddWithValue("fecaper",
DateTime.Now.Date);
            cmd.Parameters.AddWithValue("abierto",
true);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Actualizar(LibroSacramento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdateLibro";

```

```

        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("num",
e.Numero);
        cmd.Parameters.AddWithValue("max",
e.MaximoFolios);
        cmd.Parameters.AddWithValue("numfolio",
e.NumeroPorFolio);
        cmd.Parameters.AddWithValue("id",
e.IdLibro);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool CambiarEst(LibroSacramento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = (e.Abrierto) ?
"pa_CambiarEstLibro1" : "pa_CambiarEstLibro2";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            if (e.Abrierto)
            {
                cmd.Parameters.AddWithValue("fec",
DBNull.Value);
            }
            else
            {
                cmd.Parameters.AddWithValue("fec",
DateTime.Now.Date);
            }
            cmd.Parameters.AddWithValue("est",
e.Abrierto);
            cmd.Parameters.AddWithValue("id",
e.IdLibro);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

```

```

        static bool Existe(LibroSacramento e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteLibro";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
e.IdLibro);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool ExisteLS(LibroSacramento e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteLSLibro";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("tip",
e.Tipo);
                cmd.Parameters.AddWithValue("num",
e.Numero);
                cmd.Parameters.AddWithValue("id",
e.IdLibro);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool Comp(LibroSacramento e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {

```

```

        try
        {
            string sql = "pa_CompLibro";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdLibro);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
dr.Read();
            int max = Convert.ToInt32(dr[0]), nump =
Convert.ToInt32(dr[1]),
                numreg = Convert.ToInt32(dr[2]);
            return (max * nump == numreg) ? true :
false;
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

//Metodos Publicos
public static bool Guardar(LibroSacramento e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

public static void Comprobar(LibroSacramento e)
{
    if (Comp(e))
    {
        CambiarEst(new LibroSacramento(e.IdLibro,
false));
        MessageBox.Show("El libro se cerrará
automáticamente por haber\ncedido al número máximo de
registros",
                    "Libro Lleno", MessageBoxButtons.OK,
MessageBoxIcon.Information);
    }
}

public static List<LibroSacramento>
Lista(LibroSacramento e)
{
    return Secuencia(e);
}

public static List<LibroSacramento>
Lista2(LibroSacramento e)
{

```

```
        return Secuencia2(e);
    }

    public static LibroSacramento
DatosLibro(LibroSacramento e)
{
    return Secuencia3(e);
}

public static bool ExisteLibro(LibroSacramento e)
{
    return ExisteLS(e);
}

public static bool CambiarEstado(LibroSacramento e)
{
    return CambiarEst(e);
}

public static bool HayLibroAbierto(LibroSacramento e)
{
    return LibroOpen(e);
}
}
```

- ADNLugar

```
using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNLugar
    {
        static List<Lugar> Secuencia(Lugar e)
        {
            List<Lugar> lista = new List<Lugar>();
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = (e.IdLugar > 0) ?
"pa_SecuenciaLugar1" : "pa_SecuenciaLugar2";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    if (e.IdLugar > 0 == false)
                    {
                        cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    }
                    cn.Open();
                    MySqlDataReader dr = cmd.ExecuteReader();
                    while (dr.Read())
                    {
                        Lugar l = new Lugar();
                        l.IdLugar = dr.GetInt32(0);
                        l.Nombre = dr.GetString(1);
                        l.Tipo = dr.GetString(2);
                        l.Latitud = dr.GetDouble(3);
                        l.Longitud = dr.GetDouble(4);
                        lista.Add(l);
                    }
                }
            }
        }
    }
}
```

```

        l.IdLugar =
Convert.ToInt32(dr["IdLugar"]);
        l.Nombre =
Convert.ToString(dr["Nombre"]);
        if (e.IdLugar > 0 == false)
        {
            l.Direccion =
Convert.ToString(dr["Direccion"]);
        }
        lista.Add(l);
    }
}
catch (Exception ex)
{
    MessageBox.Show(ex.ToString()); return
null;
}
return lista;
}

static string direc(Lugar e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DireccionLugar";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdLugar);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            if (dr.Read())
            {
                return
Convert.ToString(dr["Direccion"]);
            }
            else
            {
                return "";
            }
        }
        catch (Exception ex)
{
    MessageBox.Show(ex.ToString()); return
"";
}
    }
}

static bool Nuevo(Lugar e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
{

```

```

        string sql = "pa_InsertarLugar";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("nom",
e.Nombre);
        cmd.Parameters.AddWithValue("dir",
e.Direccion);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Actualizar(Lugar e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdateLugar";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cmd.Parameters.AddWithValue("dir",
e.Direccion);

            cmd.Parameters.AddWithValue("id", e.IdLugar);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Existe(Lugar e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteLugar";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);

```

```

        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("id",
e.IdLugar);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}

static bool ExisteNom(Lugar e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteNomLugar";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cmd.Parameters.AddWithValue("id",
e.IdLugar);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Borrar(Lugar e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeleteLugar";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdLugar);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
    }
}

```

```
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

//Metodos publicos
public static List<Lugar> Lista(Lugar e)
{
    return Secuencia(e);
}

public static bool Guardar(Lugar e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

public static bool ExisteNombre(Lugar e)
{
    return ExisteNom(e);
}

public static bool Eliminar(Lugar e)
{
    return Borrar(e);
}

public static string Direccion(Lugar e)
{
    return direc(e);
}
```

- ADNMatrimonio

```
using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNMatrimonio
    {
        static Matrimonio CrearEntidad(MySqlDataReader
lector)
        {
            Matrimonio e = new Matrimonio();
            e.IdMatrimonio = Convert.ToInt32(lector[0]);
            e.Esposo = new Feligres();
            e.Esposo.IdFeligres = Convert.ToInt32(lector[1]);
        }
    }
}
```

```

        e.Esposo.NomPersona = new Persona();
        e.Esposo.NomPersona.ApePaterno =
Convert.ToString(lector[2]);
        e.Esposo.NomPersona.ApeMaterno =
Convert.ToString(lector[3]);
        e.Esposo.NomPersona.Nombres =
Convert.ToString(lector[4]);
        e.Esposo.Padre = Convert.ToString(lector[5]);
        e.Esposo.Madre = Convert.ToString(lector[6]);
        e.EstCivilEsposo = Convert.ToString(lector[7]);
        e.Esposo.NomPersona.FecNac =
Convert.ToDateTime(lector[8]);
        e.EdadEsposo = Convert.ToInt32(lector[9]);
        e.TembauEsposo = new Templo();
        e.TembauEsposo.IdTemplo =
Convert.ToInt32(lector[10]);
        e.FecBauEsposo = Convert.ToDateTime(lector[11]);
        e.Esposa = new Feligres();
        e.Esposa.IdFeligres =
Convert.ToInt32(lector[12]);
        e.Esposa.NomPersona = new Persona();
        e.Esposa.NomPersona.ApePaterno =
Convert.ToString(lector[13]);
        e.Esposa.NomPersona.ApeMaterno =
Convert.ToString(lector[14]);
        e.Esposa.NomPersona.Nombres =
Convert.ToString(lector[15]);
        e.Esposa.Padre = Convert.ToString(lector[16]);
        e.Esposa.Madre = Convert.ToString(lector[17]);
        e.EstCivilEsposa = Convert.ToString(lector[18]);
        e.Esposa.NomPersona.FecNac =
Convert.ToDateTime(lector[19]);
        e.EdadEsposa = Convert.ToInt32(lector[20]);
        e.TembauEsposa = new Templo();
        e.TembauEsposa.IdTemplo =
Convert.ToInt32(lector[21]);
        e.FecBauEsposa = Convert.ToDateTime(lector[22]);
        e.NomTemplo = new Templo();
        e.NomTemplo.IdTemplo =
Convert.ToInt32(lector[23]);
        e.NomTemplo.Nombre =
Convert.ToString(lector[24]);
        e.NomTemplo.LugarPro = new
Provincia(Convert.ToString(lector[25]));
        e.NomTemplo.Distrito =
Convert.ToString(lector[26]);
        e.Fecha = Convert.ToDateTime(lector[27]);
        e.Ministro = Convert.ToString(lector[28]);
        e.Padrino = Convert.ToString(lector[29]);
        e.Madrina = Convert.ToString(lector[30]);
        e.Testigo1 = Convert.ToString(lector[31]);
        e.Testigo2 = Convert.ToString(lector[32]);
        e.Libro = new LibroSacramento();
        e.Libro.IdLibro = Convert.ToInt32(lector[33]);
        e.Libro.Numero = Convert.ToInt32(lector[34]);
        e.Folio = Convert.ToInt32(lector[35]);
        e.Numero = Convert.ToInt32(lector[36]);
        e.Anotaciones = Convert.ToString(lector[37]);
        e.Libro.Abrierto = Convert.ToBoolean(lector[38]);
    return e;
}

```

```

        static List<Matrimonio> Secuencia(Persona e1, Persona
e2)
    {
        List<Matrimonio> lista = new List<Matrimonio>();
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_SecuenciaMatrimonio";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("apep1",
e1.ApePaterno);
                cmd.Parameters.AddWithValue("apem1",
e1.ApeMaterno);
                cmd.Parameters.AddWithValue("apep2",
e2.ApePaterno);
                cmd.Parameters.AddWithValue("apem2",
e2.ApeMaterno);
                cn.Open();
                MySqlDataReader dr = cmd.ExecuteReader();
                while (dr.Read())
                {
                    lista.Add(CrearEntidad(dr));
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
null;
            }
        }
        return lista;
    }

    static Matrimonio CrearEntidad2(MySqlDataReader
lector)
{
    Matrimonio e = new Matrimonio();
    e.Folio = Convert.ToInt32(lector["Folio"]);
    e.Numero = Convert.ToInt32(lector["Numero"]);
    return e;
}

    static Matrimonio Secuencia2(LibroSacramento e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_SecuenciaMatrimonio2";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdLibro);
            cn.Open();

```

```

        MySqlDataReader dr = cmd.ExecuteReader();
        if (dr.Read())
        {
            return CrearEntidad2(dr);
        }
        else
        {
            return null;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
    null;
}
}

static Matrimonio CrearEntidad3(MySqlDataReader
lector)
{
    Matrimonio e = new Matrimonio();
    e.Esposo = new Feligres();
    e.Esposo.IdFeligres = Convert.ToInt32(lector[0]);
    e.Esposo.NomPersona = new Persona();
    e.Esposo.NomPersona.ApePaterno =
    Convert.ToString(lector[1]);
    e.Esposo.NomPersona.ApeMaterno =
    Convert.ToString(lector[2]);
    e.Esposo.NomPersona.Nombres =
    Convert.ToString(lector[3]);
    if (Convert.ToInt32(lector[4]) == 27)
    {
        e.Esposo.LugNacDepa = new
Departamento(Convert.ToString(lector[6]),
            new Pais(Convert.ToInt32(lector[4])));
        e.Esposo.LugNacPro = new
Provincia(Provincia(e.Esposo.IdFeligres));
    }
    else
    {
        e.Esposo.LugNacDepa = new
Departamento(Convert.ToString(lector[5]),
            new Pais(Convert.ToInt32(lector[4])));
        e.Esposo.LugNacPro = new
Provincia(Convert.ToString(lector[6]));
    }
    e.Esposa = new Feligres();
    e.Esposa.IdFeligres = Convert.ToInt32(lector[7]);
    e.Esposa.NomPersona = new Persona();
    e.Esposa.NomPersona.ApePaterno =
    Convert.ToString(lector[8]);
    e.Esposa.NomPersona.ApeMaterno =
    Convert.ToString(lector[9]);
    e.Esposa.NomPersona.Nombres =
    Convert.ToString(lector[10]);
    if (Convert.ToInt32(lector[11]) == 27)
    {
        e.Esposa.LugNacDepa = new
Departamento(Convert.ToString(lector[13]),
            new Pais(Convert.ToInt32(lector[11])));
    }
}

```

```

        e.Esposa.LugNacPro = new
Provincia(Provincia(e.Esposa.IdFeligres));
    }
    else
    {
        e.Esposa.LugNacDepa = new
Departamento(Convert.ToString(lector[12]),
            new Pais(Convert.ToInt32(lector[11])));
        e.Esposa.LugNacPro = new
Provincia(Convert.ToString(lector[13]));
    }
    e.NomTemplo = new Templo();
    e.NomTemplo.Nombre =
Convert.ToString(lector[14]);
    e.NomTemplo.Distrito =
Convert.ToString(lector[15]);
    e.Fecha = Convert.ToDateTime(lector[16]);
    e.Ministro = Convert.ToString(lector[17]);
    e.Padrino = Convert.ToString(lector[18]);
    e.Madrina = Convert.ToString(lector[19]);
    e.Testigo1 = Convert.ToString(lector[20]);
    e.Testigo2 = Convert.ToString(lector[21]);
    e.Libro = new LibroSacramento();
    e.Libro.Numero = Convert.ToInt32(lector[22]);
    e.Folio = Convert.ToInt32(lector[23]);
    e.Numero = Convert.ToInt32(lector[24]);
    e.Anotaciones = Convert.ToString(lector[25]);
    return e;
}

static Matrimonio Secuencia3(Feligres e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_SecuenciaMatrimonio3";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idf",
e.IdFeligres);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            if (dr.Read())
            {
                return CrearEntidad3(dr);
            }
            else
            {
                return null;
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
null;
        }
    }
}

```

```

        static string Provincia(int idfeligres)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ProvinciaFeligres1";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
idfeligres);
                cn.Open();
                MySqlDataReader dr = cmd.ExecuteReader();
                dr.Read();
                return Convert.ToString(dr[0]);
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
null;
            }
        }
    }

    static bool Nuevo(Matrimonio e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_InsertarMatrimonio";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("espo",
e.Esposo.IdFeligres);
                cmd.Parameters.AddWithValue("esto",
e.EstCivilEsposo);
                cmd.Parameters.AddWithValue("edado",
e.EdadEsposo);
                cmd.Parameters.AddWithValue("tambauo",
e.TemBauEsposo.IdTemplo);
                cmd.Parameters.AddWithValue("fecbauo",
e.FecBauEsposo);
                cmd.Parameters.AddWithValue("espa",
e.Esposa.IdFeligres);
                cmd.Parameters.AddWithValue("esta",
e.EstCivilEsposa);
                cmd.Parameters.AddWithValue("edada",
e.EdadEsposa);
                cmd.Parameters.AddWithValue("tambaua",
e.TemBauEsposa.IdTemplo);
                cmd.Parameters.AddWithValue("fecbaua",
e.FecBauEsposa);
                cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
            }
        }
    }
}

```

```

e.Fecha);
cmd.Parameters.AddWithValue("fec",
e.Ministro);
cmd.Parameters.AddWithValue("min",
e.Padrino);
cmd.Parameters.AddWithValue("pad",
e.Madrina);
cmd.Parameters.AddWithValue("mad",
e.Testigo1);
cmd.Parameters.AddWithValue("test1",
e.Testigo2);
cmd.Parameters.AddWithValue("test2",
e.Libro.IdLibro);
cmd.Parameters.AddWithValue("idl",
e.Folio);
cmd.Parameters.AddWithValue("fol",
e.Numero);
cmd.Parameters.AddWithValue("num",
e.Anotaciones);
cn.Open();
return
Convert.ToBoolean(cmd.ExecuteNonQuery());
}
catch (Exception ex)
{
MessageBox.Show(ex.ToString()); return
false;
}
}
}
}

static bool Actualizar(Matrimonio e)
{
using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
try
{
string sql = "pa_UpdateMatrimonio";
MySqlCommand cmd = new MySqlCommand(sql,
cn);
cmd.CommandType =
CommandType.StoredProcedure;
cmd.Parameters.AddWithValue("esto",
e.EstCivilEsposo);
cmd.Parameters.AddWithValue("tambauo",
e.TemBauEsposo.IdTemplo);
cmd.Parameters.AddWithValue("fecbauo",
e.FecBauEsposo);
cmd.Parameters.AddWithValue("esta",
e.EstCivilEsposa);
cmd.Parameters.AddWithValue("tambaua",
e.TemBauEsposa.IdTemplo);
cmd.Parameters.AddWithValue("fecbaua",
e.FecBauEsposa);
cmd.Parameters.AddWithValue("idt",
e.NomTemplo.IdTemplo);
cmd.Parameters.AddWithValue("fec",
e.Fecha);
cmd.Parameters.AddWithValue("min",
e.Ministro);

```

```

                cmd.Parameters.AddWithValue("pad",
e.Padrino);
                cmd.Parameters.AddWithValue("mad",
e.Madrina);
                cmd.Parameters.AddWithValue("test1",
e.Testigo1);
                cmd.Parameters.AddWithValue("test2",
e.Testigo2);
                cmd.Parameters.AddWithValue("anot",
e.Anotaciones);
                cmd.Parameters.AddWithValue("id",
e.IdMatrimonio);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool Existe(Matrimonio e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteMatrimonio";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
e.IdMatrimonio);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool ExisteRegSac(Matrimonio e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_ExisteRegSacMatrimonio";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;

```

```

e.Libro.IdLibro); cmd.Parameters.AddWithValue("idl",
e.Folio); cmd.Parameters.AddWithValue("fol",
e.Numero); cmd.Parameters.AddWithValue("num",
e.IdMatrimonio); cmd.Parameters.AddWithValue("id",
cn.Open();
return
Convert.ToBoolean(cmd.ExecuteScalar());
}
catch (Exception ex)
{
    MessageBox.Show(ex.ToString()); return
false;
}
}

static string Fel(Matrimonio e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_FelMatrimonio";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("idl",
e.Libro.IdLibro);
        cmd.Parameters.AddWithValue("fol",
e.Folio);
        cmd.Parameters.AddWithValue("num",
e.Numero);
        cn.Open(); MySqlDataReader dr =
cmd.ExecuteReader(); dr.Read();
        return Convert.ToString(dr[0]) + " " +
Convert.ToString(dr[1]) + ", " + Convert.ToString(dr[2]);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
}

//Metodos Publicos
public static List<Matrimonio> Lista(Persona e1,
Persona e2)
{
    return Secuencia(e1,e2);
}

public static Matrimonio FolioNumero(LibroSacramento
e)
{
    return Secuencia2(e);
}

```

```

        }

        public static bool Guardar(Matrimonio e)
        {
            if (Existe(e))
            {
                return Actualizar(e);
            }
            else
            {
                if (Nuevo(e))
                {
                    ADNLibroSacramento.Comprobar(e.Libro);
                }
                else
                {
                    return false;
                }
            }
        }

        public static bool ExisteLFN(Matrimonio e)
        {
            return ExisteRegSac(e);
        }

        public static string FeligresLFN(Matrimonio e)
        {
            return Fel(e);
        }

        public static Matrimonio Datos(Feligres e)
        {
            return Secuencia3(e);
        }
    }
}

```

- ADNPais

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNPais
    {
        static Pais CrearEntidad(MySqlDataReader lector)
        {
            Pais e = new Pais();
            e.IdPais = Convert.ToInt32(lector["IdPais"]);
            e.Nombre = Convert.ToString(lector["Nombre"]);
            return e;
        }

        static List<Pais> Secuencia()
        {
            List<Pais> lista = new List<Pais>();

```

```

        using(MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_SecuenciaPais";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cn.Open();
        MySqlDataReader dr = cmd.ExecuteReader();
        while (dr.Read())
        {
            lista.Add(CrearEntidad(dr));
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
return lista;
}

static List<Pais> Secuencia2(int op, string nom)
{
    List<Pais> lista = new List<Pais>();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = (op == 0) ?
"pa_SecuenciaPais21" : "pa_SecuenciaPais22";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom", nom);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            while (dr.Read())
            {
                lista.Add(CrearEntidad(dr));
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
null;
        }
    }
    return lista;
}

static bool Nuevo(Pais e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {

```

```

        try
        {
            string sql = "pa_InsertarPais";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }

    static bool Actualizar(Pais e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdatePais";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cmd.Parameters.AddWithValue("id",
e.IdPais);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Existe(Pais e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExistePais";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;

```

```

                cmd.Parameters.AddWithValue("id",
e.IdPais);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool ExisteNom(Pais e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteNomPais";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cmd.Parameters.AddWithValue("id",
e.IdPais);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Borrar(Pais e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeletePais";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdPais);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {

```

```

                MessageBox.Show(ex.ToString()); return
        false;
    }
}

//Metodos publicos
public static List<Pais> Lista()
{
    return Secuencia();
}

public static List<Pais> Lista2(int op,string nom)
{
    return Secuencia2(op,nom);
}

public static bool Guardar(Pais e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

public static bool ExisteNombre(Pais e)
{
    return ExisteNom(e);
}

public static bool Eliminar(Pais e)
{
    return Borrar(e);
}
}
}

```

- ADNPersona

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNPersona
    {
        static int Max()
        {
            int id = 0;
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_MaxIdPersona";

```

```

        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cn.Open();
        MySqlDataReader dr = cmd.ExecuteReader();
        dr.Read(); id = Convert.ToInt32(dr[0]);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return 0;
    }
}
return id;
}

static bool Nuevo(Persona e)
{
    using(MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_InsertarPersona";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("apep",
e.ApePaterno);
            cmd.Parameters.AddWithValue("apem",
e.ApeMaterno);
            cmd.Parameters.AddWithValue("nom",
e.Nombres);
            cmd.Parameters.AddWithValue("fec",
e.FecNac);
            cmd.Parameters.AddWithValue("sexo",
e.Sexo);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Actualizar(Persona e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdatePersona";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;

```

```

                cmd.Parameters.AddWithValue("apep",
e.ApePaterno);
                cmd.Parameters.AddWithValue("apem",
e.ApeMaterno);
                cmd.Parameters.AddWithValue("nom",
e.Nombres);
                cmd.Parameters.AddWithValue("fec",
e.FecNac);
                cmd.Parameters.AddWithValue("sexo",
e.Sexo);
                cmd.Parameters.AddWithValue("id",
e.IdPersona);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool Existe(Persona e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExistePersona";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdPersona);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool ExisteNA(Persona e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteNAPersona";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;

```

```

                cmd.Parameters.AddWithValue("apep",
e.ApePaterno);
                cmd.Parameters.AddWithValue("apem",
e.ApeMaterno);
                cmd.Parameters.AddWithValue("nom",
e.Nombres);
                cmd.Parameters.AddWithValue("id",
e.IdPersona);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    static bool Borrar(Persona e)
    {
        using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_DeletePersona";
                MySqlCommand cmd = new MySqlCommand(sql,
cn);
                cmd.CommandType =
CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
e.IdPersona);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

    //Metodos Publicos
    public static bool Guardar(Persona e)
    {
        if (Existe(e))
        {
            return Actualizar(e);
        }
        else
        {
            return Nuevo(e);
        }
    }

    public static bool ExisteNomApe(Persona e)
{
    return ExisteNA(e);
}

```

```

        }

        public static int MaxId()
        {
            return Max();
        }

        public static bool Eliminar(Persona e)
        {
            return Borrar(e);
        }

    }
}

```

- ADNPersonal

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNPersonal
    {
        static Personal CrearEntidad(MySqlDataReader lector)
        {
            Personal e = new Personal();
            e.IdPersonal =
Convert.ToInt32(lector["IdPersonal"]);
            e.NomPersona = new
Personal(Convert.ToInt32(lector["IdPersona"]),
Convert.ToString(lector["ApePaterno"]),
Convert.ToString(lector["ApeMaterno"]),
Convert.ToString(lector["Nombres"]),
Convert.ToDateTime(lector["FecNac"]),
Convert.ToString(lector["Sexo"]));
            e.NomTipo = new
Tipos(Convert.ToInt32(lector["IdTipo"]),
Convert.ToString(lector["Nombre"]), "");
            e.Domicilio =
Convert.ToString(lector["Domicilio"]);
            e.NumTelefono =
Convert.ToString(lector["NumTelefono"]);
            e.NumCelular =
Convert.ToString(lector["NumCelular"]);
            return e;
        }

        static List<Personal> Secuencia(Persona p, int op)
        {
            List<Personal> lista = new List<Personal>();
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = (op == 1) ?
"pa_SecuenciaPersonal11" : "pa_SecuenciaPersonal12";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);

```

```

        cmd.CommandType =
CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("apep", p.ApePaterno);

cmd.Parameters.AddWithValue("apem", p.ApeMaterno);
cn.Open();
MySqlDataReader dr = cmd.ExecuteReader();
while (dr.Read())
{
    lista.Add(CrearEntidad(dr));
}
}
catch (Exception ex)
{
    MessageBox.Show(ex.ToString()); return
null;
}
return lista;
}

static List<Personal> Secuencia2(Persona e)
{
    List<Personal> lista = new List<Personal>();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_SecuenciaPersonal2";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("apep",
e.ApePaterno);
            cn.Open();
            MySqlDataReader dr = cmd.ExecuteReader();
            while (dr.Read())
            {

                Personal p = new Personal();
                p.IdPersonal =
Convert.ToInt32(dr[0]);
                p.NomPersona = new Persona();
                p.NomPersona.ApePaterno =
Convert.ToString(dr[1]);
                p.NomPersona.ApeMaterno =
Convert.ToString(dr[2]);
                p.NomPersona.Nombres =
Convert.ToString(dr[3]);
                p.NomPersona.FecNac =
Convert.ToDateTime(dr[4]);
                p.NomTipo = new Tipos();
                p.NomTipo.Nombre =
Convert.ToString(dr[5]);
                lista.Add(p);
            }
        }
        catch (Exception ex)
        {

```

```

        MessageBox.Show(ex.ToString()); return
    null;
    }
}
return lista;
}

static bool Nuevo(Personal p)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_InsertarPersonal";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;

            cmd.Parameters.AddWithValue("idp", p.NomPersona.IdPersona);

            cmd.Parameters.AddWithValue("idt", p.NomTipo.IdTipo);

            cmd.Parameters.AddWithValue("dom", p.Domicilio);

            cmd.Parameters.AddWithValue("numc", p.NumCelular);

            cmd.Parameters.AddWithValue("numt", p.NumTelefono);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString());
            return false;
        }
    }
}

static bool Actualizar(Personal p)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdatePersonal";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("idt",
p.NomTipo.IdTipo);
            cmd.Parameters.AddWithValue("dom",
p.Domicilio);
            cmd.Parameters.AddWithValue("numc",
p.NumCelular);
            cmd.Parameters.AddWithValue("numt",
p.NumTelefono);
        }
    }
}

```

```

        cmd.Parameters.AddWithValue("id",
p.IdPersonal);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Existe(Personal p)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExistePersonal";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
p.IdPersonal);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Borrar(Personal p)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeletePersonal";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
p.IdPersonal);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

```

```

        }
    }

    //Metodos Publicos
    public static List<Personal> Lista(Persona e,int op)
    {
        return Secuencia(e,op);
    }

    public static List<Personal> Lista2(Persona e)
    {
        return Secuencia2(e);
    }

    public static bool Guardar(Personal e)
    {
        if (Existe(e))
        {
            return Actualizar(e);
        }
        else
        {
            return Nuevo(e);
        }
    }

    public static bool Eliminar(Personal e)
    {
        if (Borrar(e))
        {
            return ADNPersona.Eliminar(new
Personal(e.NomPersona.IdPersona));
        }
        else
        {
            return false;
        }
    }
}
}

```

- ANDProvincia

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNProvincia
    {
        static Provincia CrearEntidad(MySqlDataReader
lector)
        {
            Provincia e = new Provincia();
            e.IdProvincia =
Convert.ToInt32(lector["IdProvincia"]);
            e.Nombre = Convert.ToString(lector["Nombre"]);
            return e;
        }
    }
}

```

```

        static List<Provincia> Secuencia(Departamento e)
    {
        List<Provincia> lista = new List<Provincia>();
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_SecuenciaProvincia";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("id",
                e.IdDepartamento);
                cn.Open();
                MySqlDataReader dr =
                cmd.ExecuteReader();
                while (dr.Read())
                {
                    lista.Add(CrearEntidad(dr));
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                null;
            }
        }
        return lista;
    }

    static bool Nuevo(Provincia e)
    {
        using (MySqlConnection cn = new
        MySqlConnection(Conexion.Cadena))
        {
            try
            {
                string sql = "pa_InsertarProvincia";
                MySqlCommand cmd = new MySqlCommand(sql,
                cn);
                cmd.CommandType =
                CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("nom",
                e.Nombre);
                cmd.Parameters.AddWithValue("idd",
                e.NomDepartamento.IdDepartamento);
                cn.Open();
                return
                Convert.ToBoolean(cmd.ExecuteNonQuery());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
                false;
            }
        }
    }

    static bool Actualizar(Provincia e)

```

```

        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_UpdateProvincia";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    cmd.Parameters.AddWithValue("id",
e.IdProvincia);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteNonQuery());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool Existe(Provincia e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_ExisteProvincia";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("id",
e.IdProvincia);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteScalar());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool ExisteNom(Provincia e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_ExisteNomProvincia";

```

```

                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    cmd.Parameters.AddWithValue("id",
e.IdProvincia);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteScalar());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool Borrar(Provincia e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_DeleteProvincia";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdProvincia);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

//Metodos publicos
public static List<Provincia> Lista(Departamento e)
{
    return Secuencia(e);
}

public static bool Guardar(Provincia e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

```

```

        }

        public static bool ExisteNombre(Provincia e)
        {
            return ExisteNom(e);
        }

        public static bool Eliminar(Provincia e)
        {
            return Borrar(e);
        }
    }
}

```

- ADNTemplo

```

using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNTemplo
    {
        static Templo CrearEntidad(MySqlDataReader lector)
        {
            Templo e = new Templo();
            e.IdTemplo =
Convert.ToInt32(lector["IdTemplo"]);
            e.Nombre = Convert.ToString(lector["Nombre"]);
            e.Tipo = Convert.ToString(lector["Tipo"]);
            //lector[3]=p.IdPais
            if (Convert.ToInt32(lector[3]) == 27)
            {
                e.LugarDepa = new
Departamento(Convert.ToString(lector[5]),
                new Pais(Convert.ToInt32(lector[3])));
                e.LugarPro = new
Provincia(Provincia(e.IdTemplo));
            }
            else
            {
                e.LugarDepa = new
Departamento(Convert.ToString(lector[4]),
                new Pais(Convert.ToInt32(lector[3])));
                e.LugarPro = new
Provincia(Convert.ToString(lector[5]));
            }
            e.Distrito =
Convert.ToString(lector["Distrito"]);
            return e;
        }

        static string Provincia(int IdTemplo)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {

```

```

        string sql = "pa_ProvinciaTemplo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("id",
IdTemplo);
        cn.Open();
        MySqlDataReader dr =
cmd.ExecuteReader();
        dr.Read();
        return Convert.ToString(dr[0]);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
}

static List<Templo> Secuencia(Templo e,int op)
{
    List<Templo> lista = new List<Templo>();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = (op == 1) ?
"pa_SecuenciaTemplos2" : ((op == 2) ?
"pa_SecuenciaTemplos3" :
"pa_SecuenciaTemplos1");
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cn.Open();
            MySqlDataReader dr =
cmd.ExecuteReader();
            while (dr.Read())
            {
                lista.Add(CrearEntidad(dr));
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
null;
        }
    }
    return lista;
}

static bool Nuevo(Templo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try

```

```

    {
        string sql = "pa_InsertarTemplo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("nom",
e.Nombre);
        cmd.Parameters.AddWithValue("tip",
e.Tipo);
        cmd.Parameters.AddWithValue("depa",
e.LugarDepa.IdDepartamento);
        if (e.LugarPro.IdProvincia > 0)
        {
            cmd.Parameters.AddWithValue("pro",
e.LugarPro.IdProvincia);
        }
        else
        {
            cmd.Parameters.AddWithValue("pro",
DBNull.Value);
        }
        cmd.Parameters.AddWithValue("dis",
e.Distrito);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}

static bool Actualizar(Templo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_UpdateTemplo";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            cmd.Parameters.AddWithValue("tip",
e.Tipo);
            cmd.Parameters.AddWithValue("depa",
e.LugarDepa.IdDepartamento);
            if (e.LugarPro.IdProvincia > 0)
            {
                cmd.Parameters.AddWithValue("pro",
e.LugarPro.IdProvincia);
            }
            else
            {

```

```

                cmd.Parameters.AddWithValue("pro",
DBNull.Value);
            }
            cmd.Parameters.AddWithValue("dis",
e.Distrito);
            cmd.Parameters.AddWithValue("id",
e.IdTemplo);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteNonQuery());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool Existe(Templo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = "pa_ExisteTemplo";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("id",
e.IdTemplo);
            cn.Open();
            return
Convert.ToBoolean(cmd.ExecuteScalar());
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString()); return
false;
        }
    }
}

static bool ExisteN(Templo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
    {
        try
        {
            string sql = (e.LugarPro.IdProvincia >
0) ? "pa_ExisteNTemplo1" : "pa_ExisteNTemplo2";
            MySqlCommand cmd = new MySqlCommand(sql,
cn);
            cmd.CommandType =
CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("nom",
e.Nombre);
            if (e.LugarPro.IdProvincia > 0)

```

```

        {
            cmd.Parameters.AddWithValue("pro",
e.LugarPro.IdProvincia);
        }
        else
        {
            cmd.Parameters.AddWithValue("depa",
e.LugarDepa.IdDepartamento);
        }
        cmd.Parameters.AddWithValue("id",
e.IdTemplo);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Borrar(Templo e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_DeleteTemplo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("id",
e.IdTemplo);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static Templo NomLug(Templo e)
{
    Templo t = new Templo();
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_NomLugTemplo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;

```

```

        cmd.Parameters.AddWithValue("id",
e.IdTemplo);
        cn.Open(); MySqlDataReader dr =
cmd.ExecuteReader(); dr.Read();

        t.Nombre = Convert.ToString(dr[0]);

        if (Convert.ToInt32(dr[1]) == 27)
        {

            t.LugarDepa = new
Departamento(Convert.ToString(dr[3]),
new
Pais(Convert.ToInt32(dr[1])));
            t.LugarPro = new
Provincia(Provincia(e.IdTemplo));
        }
        else
        {
            t.LugarDepa = new
Departamento(Convert.ToString(dr[2]),
new
Pais(Convert.ToInt32(dr[1])));
            t.LugarPro = new
Provincia(Convert.ToString(dr[3]));
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
null;
    }
}
return t;
}

//Metodos Publicos
public static List<Templo> Lista(Templo e, int
opcion)
{
    return Secuencia(e,opcion);
}

public static bool Guardar(Templo e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

public static bool Eliminar(Templo e)
{
    return Borrar(e);
}

public static bool ExisteNombre(Templo e)
{

```

```

        return ExisteN(e);
    }

    public static Templo NombreLugar(Templo e)
    {
        return NomLug(e);
    }
}

● ADNTipos
using Entidades;
using MySql.Data;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.Data;

namespace AccesoDatosNegocio
{
    public class ADNTipos
    {
        static Tipos CrearEntidad(MySqlDataReader lector)
        {
            Tipos e = new Tipos();
            e.IdTipo = Convert.ToInt32(lector["IdTipo"]);
            e.Nombre = Convert.ToString(lector["Nombre"]);
            return e;
        }

        static List<Tipos> Secuencia(Tipos e)
        {
            List<Tipos> lista = new List<Tipos>();
            using (MySqlConnection cn = new MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_SecuenciaTipos";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("tip",
e.Tipo);
                    cn.Open();
                    MySqlDataReader dr =
cmd.ExecuteReader();
                    while (dr.Read())
                    {
                        lista.Add(CrearEntidad(dr));
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
null;
                }
            }
            return lista;
        }

        static bool Nuevo(Tipos e)

```

```

        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_InsertarTipo";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    cmd.Parameters.AddWithValue("tip",
e.Tipo);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteNonQuery());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool Actualizar(Tipos e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {
                    string sql = "pa_UpdateTipo";
                    MySqlCommand cmd = new MySqlCommand(sql,
cn);
                    cmd.CommandType =
CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("nom",
e.Nombre);
                    cmd.Parameters.AddWithValue("id",
e.IdTipo);
                    cn.Open();
                    return
Convert.ToBoolean(cmd.ExecuteNonQuery());
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString()); return
false;
                }
            }
        }

        static bool Existe(Tipos e)
        {
            using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
            {
                try
                {

```

```

        string sql = "pa_ExisteTipo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("id",
e.IdTipo);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteScalar());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool Borrar(Tipos e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_DeleteTipo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("id",
e.IdTipo);
        cn.Open();
        return
Convert.ToBoolean(cmd.ExecuteNonQuery());
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString()); return
false;
    }
}
}

static bool ExisteNom(Tipos e)
{
    using (MySqlConnection cn = new
MySqlConnection(Conexion.Cadena))
{
    try
    {
        string sql = "pa_ExisteNomTipo";
        MySqlCommand cmd = new MySqlCommand(sql,
cn);
        cmd.CommandType =
CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("nom",
e.Nombre);
        cmd.Parameters.AddWithValue("tip",
e.Tipo);

```

```

                cmd.Parameters.AddWithValue("id",
e.IdTipo);
                cn.Open();
                return
Convert.ToBoolean(cmd.ExecuteScalar());
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString()); return
false;
            }
        }
    }

//Metodos publicos
public static List<Tipos> Lista(Tipos e)
{
    return Secuencia(e);
}

public static bool Guardar(Tipos e)
{
    if (Existe(e))
    {
        return Actualizar(e);
    }
    else
    {
        return Nuevo(e);
    }
}

public static bool Eliminar(Tipos e)
{
    return Borrar(e);
}

public static bool ExisteNombre(Tipos e)
{
    return ExisteNom(e);
}
}
}

```

- Conexión

```

namespace AccesoDatosNegocio
{
    public class Conexion
    {
        public static string Cadena
        {
            get { return
"Server=localhost;Database=dbParroquia;Uid=parroquia;Pwd=vice
ntinos2013"; }
        }
    }
}

```

**Código fuente de la capa Presentación (Ventanas Principales)**

- frmAsignacion.cs

```

using Utilidades;
using Entidades;
using AccesoDatosNegocio;

namespace Presentacion
{
    public partial class frmAsignacion :
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmAsignacion()
        {
            InitializeComponent();
        }

        static int idcomunidad = 0;

        void HabilitarNuevo(bool estado) {
            tbxComunidad.Enabled = !estado;
            lbxComunidades.Enabled = !estado;
            tbxApellidoP.Enabled = estado;
            dgvPersonal.Enabled = estado;
            btnGuardar.Enabled = estado; btnBuscar.Enabled =
!estado;
            btnAgregar.Enabled = estado; btnQuitar.Enabled =
estado;
        }

        void VerComunidades() {
            Comunidad c = new Comunidad();
            c.Nombre = tbxComunidad.Text.Trim();
            lbxComunidades.DataSource =
ADNComunidad.Lista2(c);
            lbxComunidades.DisplayMember = "Nombre";
            lbxComunidades.ValueMember = "IdComunidad";
            dgvAsignacion.Rows.Clear(); btnNuevo.Enabled =
false;
        }

        void VerPersonal() {
            dgvPersonal.Rows.Clear();
            Persona p = new Persona();
            p.ApePaterno = tbxApellidoP.Text.Trim();
            List<Personal> lista = ADNPersonal.Lista2(p);
            int con = 0;
            foreach (Personal item in lista)
            {
                for (int i = 0; i <
dgvAsignacion.Rows.Count; i++)
                {
                    if (item.IdPersonal ==
Convert.ToInt32(dgvAsignacion.Rows[i].Cells[0].Value))
                    {
                        con++; break;
                    }
                }
                if (con==0)
                {
                    dgvPersonal.Rows.Add(item.IdPersonal,
item.NomPersona.ApePaterno + " " +
item.NomPersona.ApeMaterno + ", " +
item.NomPersona.Nombres,

```

```

        item.NomPersona.FecNac.ToString("dd/MM/yyyy"),
        item.NomTipo.Nombre);
    }
    con = 0;
}
}

private void frmAsignacion_Load(object sender,
EventArgs e)
{
    idcomunidad = 0; VerComunidades();
}

private void btnNuevo_Click(object sender, EventArgs e)
{
    if (btnNuevo.Text == "Asignar")
    {
        btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        HabilitarNuevo(true); VerPersonal();
        btnQuitar.Enabled = (dgvAsignacion.RowCount
> 0) ? true : false;
    }
    else
    {
        btnNuevo.Text = "Asignar"; btnNuevo.Image =
global::Presentacion.Resource1.hand_property_icon;
        HabilitarNuevo(false); tbxComunidad.Clear();
        VerComunidades();
        tbxApellidoP.Clear();
        dgvAsignacion.Rows.Clear(); dgvPersonal.Rows.Clear();
        idcomunidad = 0;
    }
}

private void btnGuardar_Click(object sender,
EventArgs e)
{
    List<Asignacion> lista = new List<Asignacion>();
    if (dgvAsignacion.RowCount>0)
    {
        for (int i = 0; i < dgvAsignacion.RowCount;
i++)
        {
            lista.Add(new
Asignacion(idcomunidad,Convert.ToInt32(dgvAsignacion.Rows[i]
.Cells[0].Value)));
        }
    }
    else
    {
        lista.Add(new Asignacion(idcomunidad,0));
    }

    if (ADNAsignacion.Guardar(lista))
    {
        MessageBox.Show("Asignacion actualizada",
"Exito", MessageBoxButtons.OK, MessageBoxIcon.Information);
        btnNuevo.PerformClick();
    }
}

```

```

        else
        {
            MessageBox.Show("Ha ocurrido un problema",
"Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
    }

    private void btnCerrar_Click(object sender,
EventArgs e)
{
    this.Close();
}

private void lbxComunidades_Click(object sender,
EventArgs e)
{
    if (lbxComunidades.Items.Count > 0)
    {
        btnNuevo.Enabled = true;
        dgvAsignacion.Rows.Clear();
        Asignacion c = new Asignacion();
        c.IdComunidad =
Convert.ToInt32(lbxComunidades.SelectedValue);
        idcomunidad = c.IdComunidad;
        List<Personal> listaasignacion =
ADNAsignacion.Lista(c);
        foreach (Personal item in listaasignacion)
        {
            dgvAsignacion.Rows.Add(item.IdPersonal,
item.NomPersona.ApePaterno + " " +
item.NomPersona.ApeMaterno + ", "
+ item.NomPersona.Nombres,
item.NomPersona.FecNac.ToString("dd/MM/yyyy"),
item.NomTipo.Nombre);
        }
    }
    else
    {
        btnNuevo.Enabled = false;
    }
}

private void tbxComunidad_KeyPress(object sender,
KeyPressEventArgs e)
{
    if (e.KeyChar==(char)Keys.Enter)
    {
        VerComunidades();
    }
}

private void tbxApellidoP_KeyPress(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
    if (e.KeyChar == (char)Keys.Enter)
    {
        VerPersonal();
    }
}

```

```

        private void btnAgregar_Click(object sender,
EventArgs e)
{
    int con = 0;
    foreach (DataGridViewRow f in dgvPersonal.Rows)
    {
        con = (f.Selected) ? con + 1 : con;
    }
    for (int j = 0; j < con; j++)
    {
        for (int i = 0; i < dgvPersonal.RowCount;
i++)
        {
            DataGridViewRow f = dgvPersonal.Rows[i];
            if (f.Selected)
            {

dgvAsignacion.Rows.Add(f.Cells[0].Value, f.Cells[1].Value,
f.Cells[2].Value,
f.Cells[3].Value);
                dgvPersonal.Rows.RemoveAt(i); break;
            }
        }
    }
    btnAgregar.Enabled = (dgvPersonal.RowCount > 0)
? true : false;
    btnQuitar.Enabled = (dgvAsignacion.RowCount > 0)
? true : false;
}

        private void btnQuitar_Click(object sender,
EventArgs e)
{
    int con = 0;
    foreach (DataGridViewRow f in
dgvAsignacion.Rows)
    {
        con = (f.Selected) ? con + 1 : con;
    }
    for (int j = 0; j < con; j++)
    {
        for (int i = 0; i < dgvAsignacion.RowCount;
i++)
        {
            if (dgvAsignacion.Rows[i].Selected)
            {
                dgvAsignacion.Rows.RemoveAt(i);
break;
            }
        }
    }
    tbxApellidoP.Clear(); VerPersonal();
    btnQuitar.Enabled = (dgvAsignacion.RowCount > 0)
? true : false;
    btnAgregar.Enabled = (dgvPersonal.RowCount > 0)
? true : false;
}

        private void dgvPersonal_KeyPress(object sender,
KeyPressEventArgs e)
{
    btnAgregar.PerformClick();
}

```

```

        }

        private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerComunidades();
}
}

● frmBautismo.cs
using Entidades;
using AccesoDatosNegocio;
using Utilidades;

namespace Presentacion
{
    public partial class frmBautismo :
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmBautismo()
        {
            InitializeComponent();
        }

        static Bautismo bautismo = null;
        static LibroSacramento libro = null;
        static int idB = 0;
        static bool mod = false;

        public static Feligres fel = new Feligres();
        public static Templo templo = new Templo();
        public static bool invoca = false;

        void VerDatos() {
            dgvLista.Rows.Clear();
            Persona per = new Persona();
            per.ApePaterno = tbxFilApeP.Text.Trim();
            per.ApeMaterno = tbxFilApeM.Text.Trim();
            List<Bautismo> lista = ADNBautismo.Lista(per);
            foreach (var ele in lista)
            {
                dgvLista.Rows.Add(ele.DatosToArray());
            }
            btnModificar.Enabled = false;
        }

        void CalcularFolioNumero() {
            int num = 0, folio=0;
            LibroSacramento lib = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
            bautismo = ADNBautismo.FolioNumero(lib);
            if (bautismo==null ||
chbxContinuar.Checked==false)
            {
                num = 1; folio = 1;
            }
            else
            {
                num = bautismo.Numero + 1;
            }
        }
    }
}

```

```

            folio = (bautismo.Numero %
libro.NumeroPorFolio == 0) ? bautismo.Folio + 1 :
bautismo.Folio;
        }
        nudFolio.Value = new decimal(new int[] { folio,
0, 0, 0 });
        nudNumero.Value = new decimal(new int[] { num,
0, 0, 0 });
        CalcularMaxMinNumero();
    }

    void CalcularMaxMinNumero() {
        int fol = Convert.ToInt32(nudFolio.Value);
        nudNumero.Minimum = new decimal(new int[] {
((fol - 1) * libro.NumeroPorFolio) + 1, 0, 0, 0 });
        nudNumero.Maximum = new decimal(new int[] { fol
* libro.NumeroPorFolio, 0, 0, 0 });
    }

    void VerLibros(int num) {
        cbxLibro.ValueMember = "IdLibro";
        cbxLibro.DataSource = (num>0) ?
ADNLibroSacramento.Lista2(new LibroSacramento(num))
: ADNLibroSacramento.Lista2(new
LibroSacramento("Bautismo"));
        cbxLibro.DisplayMember = "Numero";
    }

    void HabilitarConMod(bool est) {
        cbxLibro.Enabled = est; nudFolio.Enabled = est;
        nudNumero.Enabled = est; chbxContinuar.Enabled =
est;
        btnBuscarFeligr.es.Enabled = est;
    }

    private void frmBautismo_Load(object sender,
EventArgs e)
{
    Metodos.CerrarForm<frmLibroSacramento>();
    VerLibros(0); mod = false;
    dtpFecha.MaxDate = DateTime.Today; idB = 0;
    fel.IdFeligr.es = 0; templo.IdTemplo=0;
}

    private void btnCerrar_Click(object sender,
EventArgs e)
{
    this.Close();
}

    private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}

    private void btnNuevo_Click(object sender, EventArgs
e)
{
    if (btnNuevo.Text == "Nuevo")
{

```

```

        btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        tabControl1.SelectedIndex = 1;
btnModificar.Enabled = false;

        if (mod==false) { CalcularFolioNumero(); }

        Metodos.Limpiar(groupBox1);
Metodos.Habilitar(groupBox1, false);
        dtpFecha.MinDate = new System.DateTime(1753,
1, 1, 0, 0, 0);
        templo.IdTemplo = 1; tbxTemplo.Text = "San
Clemente de Pisco, Ica";
        }
        else
        {
            btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
            Metodos.Limpiar(groupBox3);
Metodos.Limpiar(groupBox4);
            Metodos.Habilitar(groupBox1, true);
tabControl1.SelectedIndex = 0;
            idB = 0; btnBuscarFeligrés.Enabled = true;
            if (mod) {
                VerLibros(0); mod = false;
HabilitarConMod(true);
            }
        }
    }

private void btnGuardar_Click(object sender,
EventArgs e)
{
    if (fel.IdFeligrés > 0 && tbxMinistro.TextLength
> 0 && tbxPadrino.TextLength > 0
&& tbxMadrina.TextLength > 0 &&
templo.IdTemplo > 0)
    {
        Bautismo bau = new Bautismo();
bau.IdBautismo = idB;
        bau.Libro = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
        bau.Folio = Convert.ToInt32(nudFolio.Value);
        bau.Numero =
Convert.ToInt32(nudNúmero.Value);
        if (ADNBautismo.ExisteLFN(bau))
        {
            string nom =
ADNBautismo.FeligrésLFN(bau);
            MessageBox.Show("Ya existe un registro
en el libro, folio y número.\nFeligrés: " + nom, "Registro
Parroquial existente",
                MessageBoxButtons.OK,
MessageBoxIcon.Stop);
        }
        else
        {
            DialogResult res = (idUser == 0) ?
MessageBox.Show("¿El libro, folio y número son
correctos?.\n" +

```

```

                "Estos valores no se podran
                modificar mas adelante", "Verificacion del Registro
                Parroquial",
                MessageBoxButtons.YesNo,
                MessageBoxIcon.Question)
                : DialogResult.Yes;
            if (res == DialogResult.Yes)
            {
                bau.NomFeligres = fel;
                bau.NomTemplo = templo;
                bau.Fecha = dtpFecha.Value;
                bau.Ministro =
                tbxMinistro.Text.Trim();
                bau.Padrino =
                tbxPadrino.Text.Trim();
                bau.Madrina =
                tbxMadrina.Text.Trim();
                bau.NotaMarginal =
                tbxNota.Text.Trim();
                if (ADNBautismo.Guardar(bau))
                {
                    string r = (idB == 0) ?
                    "guardados" : "actualizados";
                    MessageBox.Show("Datos " + r + "
                    con exito.", "Exito", MessageBoxButtons.OK,
                    MessageBoxIcon.Information);
                    btnNuevo.PerformClick();
                }
                VerDatos();
            }
            Metodos.CambiarSize<frmMatrimonio>();
            Metodos.CambiarSize<frmConfirmacion>();
        }
        else
        {
            MessageBox.Show("Ha ocurrido un
            problema.", "Error", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
        }
    }
}
else
{
    MessageBox.Show("Faltan Datos", "Datos",
    MessageBoxButtons.OK, MessageBoxIcon.Stop);
}
}

private void btnBuscarFeligres_Click(object sender,
EventArgs e)
{
    fel.IdFeligres = 0; invoca = true;
    frmBuscarFeligres f = new frmBuscarFeligres();
    f.ShowDialog();
    if (fel.IdFeligres > 0)
    {
        tbxFeligres.Text = fel.NomPersona.ApePaterno
        + " " + fel.NomPersona.ApeMaterno + ", " +
                    fel.NomPersona.Nombres;
        string aux = (fel.DistritoFel.Length>0) ? "
        - "+fel.DistritoFel : "";
    }
}

```

```

        tbxLugNac.Text = fel.LugNacDepa.Nombre + " - "
        " + fel.LugNacPro.Nombre + aux;
        tbxFecNac.Text =
fel.NomPersona.FecNac.ToShortDateString();
        tbxPadre.Text = fel.Padre; tbxMadre.Text =
fel.Madre;
        if (fel.NatPadreDepa.Nombre!="")
        {
            aux = (fel.DistritoPadre.Length > 0) ? " "
- " + fel.DistritoPadre : "";
            tbxNatPadre.Text =
fel.NatPadreDepa.Nombre + " - " + fel.NatPadrePro.Nombre +
aux;
        }
        aux = (fel.DistritoMadre.Length > 0) ? " - "
+ fel.DistritoMadre : "";
        tbxNatMadre.Text = fel.NatMadreDepa.Nombre +
" - " + fel.NatMadrePro.Nombre+aux;
        DateTime fecnac = fel.NomPersona.FecNac;
        dtpFecha.MinDate = new
System.DateTime(fecnac.Year, fecnac.Month, fecnac.Day, 0, 0,
0, 0);
    }
    else
    {
        fel.IdFeligres = 0;
Metodos.Limpiar(groupBox3);
        dtpFecha.Value = DateTime.Today;
    }
    invoca = false;
}

private void btnModificar_Click(object sender,
EventArgs e)
{
    mod = true;
    DataGridViewRow f = dgvLista.CurrentRow;
    idB = Convert.ToInt32(f.Cells[0].Value);
    Feligres i = new Feligres(); i.IdFeligres =
Convert.ToInt32(f.Cells[1].Value);
    fel = ADNFeligres.DatosFel(i);
    tbxFeligres.Text = f.Cells[2].Value.ToString();
    templo.IdTemplo =
Convert.ToInt32(f.Cells[3].Value);
    tbxTemplo.Text = f.Cells[4].Value.ToString();
    dtpFecha.Value =
Convert.ToDateTime(f.Cells[5].Value);
    tbxMinistro.Text = f.Cells[6].Value.ToString();
    tbxPadrino.Text = f.Cells[7].Value.ToString();
    tbxMadrina.Text = f.Cells[8].Value.ToString();
    tbxNota.Text = f.Cells[13].Value.ToString();
    if (f.Cells[14].Value.ToString() == "False")
    {

VerLibros(Convert.ToInt32(f.Cells[9].Value));
    }
    cbxLibro.SelectedValue =
Convert.ToInt32(f.Cells[9].Value);
    string aux = (fel.DistritoFel.Length > 0) ? " - "
+ fel.DistritoFel : "";
    tbxLugNac.Text = fel.LugNacDepa.Nombre + " - " +
fel.LugNacPro.Nombre + aux;
}

```

```

        tbxFecNac.Text =
fel.NomPersona.FecNac.ToShortDateString();
        tbxPadre.Text = fel.Padre; tbxMadre.Text =
fel.Madre;
        aux = (fel.DistritoPadre.Length > 0) ? " - " +
fel.DistritoPadre : "";
        tbxNatPadre.Text = fel.NatPadreDepa.Nombre + " -
" + fel.NatPadrePro.Nombre + aux;
        aux = (fel.DistritoMadre.Length > 0) ? " - " +
fel.DistritoMadre : "";
        tbxNatMadre.Text = fel.NatMadreDepa.Nombre + " -
" + fel.NatMadrePro.Nombre + aux;

        btnNuevo.PerformClick();
        dtpFecha.MinDate = fel.NomPersona.FecNac;

        nudFolio.Value = new decimal(new int[] {
Convert.ToInt32(f.Cells[10].Value), 0, 0, 0 });
        nudNumero.Value = new decimal(new int[] {
Convert.ToInt32(f.Cells[11].Value), 0, 0, 0 });
        HabilitarConMod(false);
    }

    private void btnBuscarTemplo_Click(object sender,
EventArgs e)
{
    templo.IdTemplo = 0; invoca = true;
    frmBuscarTemplo f = new frmBuscarTemplo();
f.ShowDialog();
    if (templo.IdTemplo > 0)
    {
        tbxTemplo.Text = templo.Nombre + " de " +
templo.Tipo;
    }
    else
    {
        tbxTemplo.Clear();
    }
    invoca = false;
}

    private void tabControl1_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        tabControl1.SelectedIndex = 0;
    }
    else
    {
        tabControl1.SelectedIndex = 1;
    }
}

    private void cbxLibro_SelectedIndexChanged(object
sender, EventArgs e)
{
    LibroSacramento lib = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
    libro = ADNLibroSacramento.DatosLibro(lib);
    nudFolio.Maximum = new decimal(new int[] {
libro.MaximoFolios, 0, 0, 0 });
}

```

```

        if (mod==false)
        {
            CalcularFolioNumero();
        }
    }

    private void chbxContinuar_CheckedChanged(object
sender, EventArgs e)
{
    CalcularFolioNumero();
}

private void nudFolio_ValueChanged(object sender,
EventArgs e)
{
    CalcularMaxMinNumero();
}

private void ManejadorLetras(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
}

private void dgvLista_Click(object sender, EventArgs
e)
{
    btnModificar.Enabled=(dgvLista.Rows.Count>0) ?
true : false;
}

private void tbxMinistro_KeyPress(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
    if (Char.IsPunctuation(e.KeyChar))
    { e.Handled = false; }
}

private void ManejadorFiltro(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
    if (e.KeyChar==(char)Keys.Enter)
    {
        VerDatos();
    }
}
}
}

```

- frmComunidades.cs

```

using Entidades;
using AccesoDatosNegocio;
using Utilidades;

namespace Presentacion
{
    public partial class frmComunidades :
Presentacion.Plantillas.frmPlantillaForm

```

```

    {
        public frmComunidades()
        {
            InitializeComponent();
        }

        static bool colocar = false;
        static int idC = 0;

        void VerDatos() {
            dgvLista.Rows.Clear();
            Comunidad c = new Comunidad();
            c.Nombre = tbxFil.Text.Trim();
            List<Comunidad> lista = ADNComunidad.Lista(c);
            foreach (Comunidad item in lista)
            {
                dgvLista.Rows.Add(item.DatosToArray());
            }
            btnModificar.Enabled = false;
            btnEliminar.Enabled = false;
        }

        void VerTipos() {
            cbxTipo.DataSource = ADNTipos.Lista(new
Tipos("Comunidad"));
            cbxTipo.DisplayMember = "Nombre";
            cbxTipo.ValueMember = "IdTipo";
        }

        void VerLugares() {
            cbxLugar.DataSource = ADNLugar.Lista(new
Lugar(1));
            cbxLugar.DisplayMember = "Nombre";
            cbxLugar.ValueMember = "IdLugar";
        }

        private void frmComunidades_Load(object sender,
EventArgs e)
        {
            this.ClientSize = new System.Drawing.Size(429,
287);
            VerTipos(); cbxDia.SelectedIndex = 0;
            colocar = false; VerLugares();
            cbxLugar.SelectedIndex = 1;
            colocar = true; cbxLugar.SelectedIndex = 0; idC
= 0;
            VerDatos();
        }

        private void btnCerrar_Click(object sender,
EventArgs e)
        {
            this.Close();
        }

        private void btnNuevo_Click(object sender, EventArgs
e)
        {
            if (btnNuevo.Text == "Nuevo")
            {
                btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        }
    }
}

```

```

                btnModificar.Enabled = false;
btnEliminar.Enabled = false;
            Metodos.Limpiar(groupBox1);
Metodos.Habilitar(groupBox1, false);
            this.ClientSize = new
System.Drawing.Size(429, 500);
            dgvLista.Enabled = false;
        }
        else
        {
            btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
            Metodos.Limpiar(groupBox3);
Metodos.Habilitar(groupBox1, true);
            this.ClientSize = new
System.Drawing.Size(429, 287);
            dgvLista.Enabled = true; idC = 0;
        }
    }

private void btnGuardar_Click(object sender,
EventArgs e)
{
    if (tbxNombre.TextLength>0)
    {
        DialogResult res = (idC == 0) ?
MessageBox.Show("¿Los datos ingresados son correctos?", 
                "Verificacion del Registro",
                MessageBoxButtons.YesNo, MessageBoxIcon.Question)
                : DialogResult.Yes;
        if (res==DialogResult.Yes)
        {
            Comunidad c = new Comunidad();
            c.IdComunidad = idC;
            c.Nombre =
Metodos.LetraTipoOracion(tbxNombre.Text.Trim());
            c.Descripcion =
tbxDireccion.Text.Trim();
            c.DiaReunion = cbxDia.Text;
            c.NomTipo = new
Tipos(Convert.ToInt32(cbxTipo.SelectedValue));
            c.Lugar = new
Lugar(Convert.ToInt32(cbxLugar.SelectedValue));
            if (ADNComunidad.Guardar(c))
            {
                string r = (idC == 0) ? "guardados"
: "actualizados";
                MessageBox.Show("Datos " + r + "
correctamente.", "Datos", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                btnNuevo.PerformClick(); VerDatos();
            }
            else
            {
                MessageBox.Show("Ha ocurrido un
problema", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            }
        }
    }
}

```

```

        {
            MessageBox.Show("Debe escribir el nombre",
"Faltan Datos", MessageBoxButtons.OK, MessageBoxIcon.Stop);
        }
    }

    private void btnModificar_Click(object sender,
EventArgs e)
{
    DataGridViewRow f = dgvLista.CurrentRow;
    idC = Convert.ToInt32(f.Cells[0].Value);
    tbxNombre.Text = f.Cells[1].Value.ToString();
    tbxDescripcion.Text =
f.Cells[2].Value.ToString();
    cbxTipo.Text = f.Cells[3].Value.ToString();
    cbxLugar.Text = f.Cells[4].Value.ToString();
    tbxDireccion.Text = f.Cells[5].Value.ToString();
    cbxDia.Text = f.Cells[6].Value.ToString();
    btnNuevo.PerformClick();
}

private void btnEliminar_Click(object sender,
EventArgs e)
{
    DialogResult res = MessageBox.Show("¿Esta seguro
que desea eliminar el registro?", "Confirmacion",
MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
    if (res==DialogResult.Yes)
    {
        Comunidad c = new Comunidad();
        c.IdComunidad =
Convert.ToInt32(dgvLista.CurrentRow.Cells[0].Value);
        if (ADNComunidad.Eliminar(c))
        {
            MessageBox.Show("Datos eliminados con
exito", "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            VerDatos();
        }
        else
        {
            MessageBox.Show("Ha ocurrido un
problema", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
    }
}

private void cbxLugar_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (colocar)
    {
        tbxDireccion.Text=ADNLugar.Direccion(new
Lugar(Convert.ToInt32(cbxLugar.SelectedValue)));
    }
}

private void tbxFil_KeyPress(object sender,
KeyPressEventArgs e)
{
}

```

```

        Metodos.SoloLetras(e);
        if (e.KeyChar == (char)Keys.Enter)
        {
            VerDatos();
        }
    }

    private void dgvLista_Click(object sender, EventArgs
e)
{
    if (dgvLista.RowCount>0)
    {
        btnEliminar.Enabled = true;
        btnModificar.Enabled = true;
    }
    else
    {
        btnEliminar.Enabled = false;
        btnModificar.Enabled = false;
    }
}

private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}
}
}

```

- frmConfirmacion.cs

```

using Entidades;
using AccesoDatosNegocio;
using Utilidades;

namespace Presentacion
{
    public partial class frmConfirmacion :
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmConfirmacion()
        {
            InitializeComponent();
        }

        static int idC = 0;
        static bool mod = false;

        public static bool invoca = false;
        public static bool tembau = false;
        public static bool tem = false;
        public static Feligres fel = new Feligres();
        public static Templo templobau = new Templo();
        public static Templo templo = new Templo();

        void VerDatos() {
            dgvLista.Rows.Clear();
            Persona per = new Persona();
            per.ApePaterno = tbxFilApeP.Text.Trim();
            per.ApeMaterno = tbxFilApeM.Text.Trim();
            List<Confirmacion> lista =
ADNConfirmacion.Lista(per);

```

```

        foreach (var ele in lista)
        {
            dgvLista.Rows.Add(ele.DatosToArray());
        }
        btnModificar.Enabled = false;
    }

    void VerLibros(int num)
    {
        cbxLibro.ValueMember = "IdLibro";
        cbxLibro.DataSource = (num > 0) ?
ADNLibroSacramento.Lista2(new LibroSacramento(num))
        : ADNLibroSacramento.Lista2(new
LibroSacramento("Confirmacion"));
        cbxLibro.DisplayMember = "Numero";
    }

    void HabilitarConMod(bool est)
    {
        cbxLibro.Enabled = est; nudFolio.Enabled = est;
        nudNumero.Enabled = est;
        btnBuscarFeligr.es.Enabled = est;
    }

    private void frmConfirmacion_Load(object sender,
EventArgs e)
{
    Metodos.CerrarForm<frmLibroSacramento>();
    VerLibros(0);
    idC = 0; mod = false; templo.IdTemplo = 0;
    templobau.IdTemplo = 0; fel.IdFeligr.es = 0;
    dtpFechaBau.MaxDate = DateTime.Today;
    dtpFecha.MaxDate = DateTime.Today;
}

    private void ManejadorNumeros(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloNumeros(e);
}

    private void ManejadorLetas(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
}

    private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}

    private void btnNuevo_Click(object sender, EventArgs
e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        tabControl1.SelectedIndex = 1;
        btnModificar.Enabled = false;
    }
}

```

```

        Metodos.Limpiar(groupBox1);
        Metodos.Habilitar(groupBox1, false);
    }
    else
    {
        btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
        tabControl1.SelectedIndex = 0; idC = 0;
        Metodos.Limpiar(groupBox3);
        Metodos.Limpiar(groupBox4); Metodos.Limpiar(groupBox5);
        btnBuscarFeligres.Enabled = true;
        Metodos.Habilitar(groupBox5, false);
        Metodos.Habilitar(groupBox1, true);
        if (mod)
        {
            HabilitarConMod(true);
            VerLibros(0); mod = false;
        }
    }
}

private void btnGuardar_Click(object sender,
EventArgs e)
{
    if (fel.IdFeligres > 0 && templo.IdTemplo > 0 &&
templobau.IdTemplo > 0 && tbxLibroBautismo.TextLength > 0 &&
tbxFolioBautismo.TextLength > 0 &&
tbxMinistro.TextLength > 0 && tbxPadrinoMadrina.TextLength >
0)
    {
        Confirmacion con = new Confirmacion();
        con.IdConfirmacion = idC;
        con.Libro = new
LibroSacramento1(Convert.ToInt32(cbxLibro.SelectedValue));
        con.Folio = Convert.ToInt32(nudFolio.Value);
        con.Numero =
Convert.ToInt32(nudNumero.Value);
        if (ADNConfirmacion.ExisteLFN(con))
        {
            string nom =
ADNConfirmacion.FeligresLFN(con);
            MessageBox.Show("Ya existe un registro
en el libro, folio y número.\nFeligres: " + nom, "Registro
Parroquial existente",
                MessageBoxButtons.OK,
                MessageBoxIcon.Stop);
        }
        else
        {
            DialogResult res = (idC == 0) ?
MessageBox.Show("¿El libro, folio y número son
correctos?.\n" +
                    "Estos valores no se podran
modificar mas adelante", "Verificacion del Registro
Parroquial",
                MessageBoxButtons.YesNo,
                MessageBoxIcon.Question) : DialogResult.Yes;
            if (res == DialogResult.Yes)
            {
                con.NomFeligres = fel;
                con.NomTemplo = templo;
                con.TemploBau = templobau;

```



```

        {
            MessageBox.Show("Faltan Datos", "Datos",
                MessageBoxButtons.OK, MessageBoxIcon.Stop);
        }
    }

    private void btnModificar_Click(object sender,
EventArgs e)
{
    mod = true;
    DataGridViewRow f = dgvLista.CurrentRow;
    idC = Convert.ToInt32(f.Cells[0].Value);
    fel.IdFeligres =
Convert.ToInt32(f.Cells[1].Value);
    tbxFeligres.Text = f.Cells[2].Value.ToString();
    Feligres aux1 = ADNFeligres.LugarNac(fel);

    tbxLugNac.Text = aux1.LugNacDepa.Nombre + " - "
+ aux1.LugNacPro.Nombre;
    tbxFecNac.Text = f.Cells[3].Value.ToString();
    tbxPadre.Text = f.Cells[4].Value.ToString();
    tbxMadre.Text = f.Cells[5].Value.ToString();

    templo.IdTemplo =
Convert.ToInt32(f.Cells[6].Value);
    tbxTemplo.Text = f.Cells[7].Value.ToString();
    templobau.IdTemplo =
Convert.ToInt32(f.Cells[8].Value);
    if (templobau.IdTemplo > 1)
    {
        tbxLibroBautismo.Enabled = true;
    tbxFolioBautismo.Enabled = true;
        dtpFechaBau.Enabled = true;
    btnBuscarParroquiaBau.Enabled = true;
    }

    Templo aux2 = ADNTemplo.NombreLugar(templobau);
    tbxParroquiaBautismo.Text = aux2.Nombre + " de "
+ aux2.LugarPro.Nombre + ", " + aux2.LugarDepa.Nombre;

    tbxLibroBautismo.Text =
f.Cells[9].Value.ToString();
    tbxFolioBautismo.Text =
f.Cells[10].Value.ToString();
    dtpFechaBau.Value =
Convert.ToDateTime(f.Cells[11].Value);
    dtpFecha.Value =
Convert.ToDateTime(f.Cells[12].Value);
    tbxMinistro.Text = f.Cells[13].Value.ToString();
    tbxPadrinoMadrina.Text =
f.Cells[14].Value.ToString();
    if (f.Cells[20].Value.ToString() == "False")
    {

        VerLibros(Convert.ToInt32(f.Cells[15].Value));
    }
    cbxLibro.SelectedValue =
Convert.ToInt32(f.Cells[15].Value);
    nudFolio.Value = new decimal(new int[] {
Convert.ToInt32(f.Cells[16].Value), 0, 0, 0 });
    nudNumero.Value = new decimal(new int[] {
Convert.ToInt32(f.Cells[17].Value), 0, 0, 0 });
}

```

```

        tbxNota.Text = f.Cells[19].Value.ToString();
        btnNuevo.PerformClick();
        HabilitarConMod(false);
    }

    private void btnCerrar_Click(object sender,
EventArgs e)
{
    this.Close();
}

private void tabControl1_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        tabControl1.SelectedIndex = 0;
    }
    else
    {
        tabControl1.SelectedIndex = 1;
    }
}

private void cbxLibro_SelectedIndexChanged(object
sender, EventArgs e)
{
    LibroSacramento lib = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
nudFolio.Maximum = new decimal(new int[] {

ADNLibroSacramento.DatosLibro(lib).MaximoFolios, 0, 0, 0
});
}

private void btnBuscarFeligres_Click(object sender,
EventArgs e)
{
    invoca = true; fel.IdFeligres = 0;
    frmBuscarFeligres f = new frmBuscarFeligres();
f.ShowDialog();
    if (fel.IdFeligres>0)
    {
        tbxFeligres.Text = fel.NomPersona.ApePaterno
+ " " + fel.NomPersona.ApeMaterno + ", " +
            fel.NomPersona.Nombres;
        string aux = (fel.DistritoFel.Length > 0) ?
" - " + fel.DistritoFel : "";
        tbxLugNac.Text = fel.LugNacDepa.Nombre + " -
" + fel.LugNacPro.Nombre + aux;
        tbxFecNac.Text =
fel.NomPersona.FecNac.ToShortDateString();
        tbxPadre.Text = fel.Padre; tbxMadre.Text =
fel.Madre;
        Metodos.Limpiar(groupBox5);
        Bautismo bau = ADNBautismo.Datos(fel,0);
        if (bau==null)
        {
            Metodos.Habilitar(groupBox5, true);
            tbxParroquiaBautismo.Enabled = false;
            DateTime fecnac = fel.NomPersona.FecNac;
        }
    }
}

```

```

        dtpFechaBau.MinDate = new
System.DateTime(fecnac.Year, fecnac.Month, fecnac.Day, 0, 0,
0, 0);
    }
    else
    {
        Metodos.Habilitar(groupBox5, false);
        templobau = bau.NomTemplo;
        tbxParroquiaBautismo.Text =
templobau.Nombre + " de "+templobau.Distrito+" Pisco";
        tbxLibroBautismo.Text =
bau.Libro.Numero.ToString();
        tbxFolioBautismo.Text =
bau.Folio.ToString();
        dtpFechaBau.MinDate = new
System.DateTime(1753, 1, 1, 0, 0, 0);
        dtpFechaBau.Value = bau.Fecha;
    }
}
else
{
    fel.IdFeligrés = 0;
    Metodos.Limpiar(groupBox3);
Metodos.Limpiar(groupBox5);
}
invoca = false;
}

private void btnBuscarParroquiaBau_Click(object
sender, EventArgs e)
{
    templobau.IdTemplo = 0; tembau = true;
    frmBuscarTemplo f = new frmBuscarTemplo();
f.ShowDialog();
    if (templobau.IdTemplo>0)
    {
        tbxParroquiaBautismo.Text = templobau.Nombre
+ " de " + templobau.Tipo;
    }
    else
    {
        tbxParroquiaBautismo.Clear();
    }
    tembau = false;
}

private void btnBuscarTemplo_Click(object sender,
EventArgs e)
{
    templo.IdTemplo = 0; tem = true;
    frmBuscarTemplo f = new frmBuscarTemplo();
f.ShowDialog();
    if (templo.IdTemplo > 0)
    {
        tbxTemplo.Text = templo.Nombre + " de " +
templo.Tipo;
    }
    else
    {
        tbxTemplo.Clear();
    }
    tem = false;
}

```

```
    }

    private void dgvLista_Click(object sender, EventArgs e)
    {
        btnModificar.Enabled = (dgvLista.Rows.Count > 0) ? true : false;
    }

    private void dtpFechaBau_ValueChanged(object sender, EventArgs e)
    {
        DateTime fec = dtpFechaBau.Value;
        dtpFecha.MinDate = new System.DateTime(fec.Year, fec.Month, fec.Day, 0, 0, 0);
    }

    private void frmConfirmacion_SizeChanged(object sender, EventArgs e)
    {
        if (btnNuevo.Text.Equals("Cancelar"))
        {
            btnNuevo.PerformClick();
        }
        VerDatos();
    }

    private void tbxMinistro_KeyPress(object sender, KeyPressEventArgs e)
    {
        Metodos.SoloLetras(e);
        if (Char.IsPunctuation(e.KeyChar))
        { e.Handled = false; }
    }

    private void ManejadoFiltro(object sender, KeyPressEventArgs e)
    {
        Metodos.SoloLetras(e);
        if (e.KeyChar == (char)Keys.Enter)
        {
            btnBuscar.PerformClick();
        }
    }
}
```

- frmFeligres.cs

```
using Utilidades;
using Entidades;
using AccesoDatosNegocio;

namespace Presentacion
{
    public partial class frmFeligres : 
Presentacion.Plantillas.frmPlantillaForm

    {

        public frmFeligres()
        {

```

```

        InitializeComponent();
    }

    static bool ColocarPro = false;
    static int idF=0;

    public static Persona per=new Persona();
    public static bool invoca = false;

    void CargarDepartamento(ComboBox Combo) {
        Pais p = new Pais(27);
        //Combo.DataSource = ADNDepartamento.Lista(p);
        Combo.DataSource = (Combo==cbxDepaFel) ?
    frmPrincipal.lis1 :
        ((Combo==cbxDepaMadre) ? frmPrincipal.lis2 :
    frmPrincipal.lis3);
        Combo.DisplayMember = "Nombre";
        Combo.ValueMember = "IdDepartamento";
    }

    void CargarPais(ComboBox Combo) {
        Combo.DisplayMember = "Nombre";
        Combo.DataSource = ADNPais.Lista();
        Combo.ValueMember = "IdPais";
    }

    void CargarProvincia(CheckBox Check,ComboBox
    ComboPro,ComboBox ComboDepa) {
        if (ColocarPro)
        {
            if (Check.Checked)
            {
                Departamento d = new Departamento();
                d.IdDepartamento =
    Convert.ToInt32(ComboDepa.SelectedValue);
                ComboPro.DataSource =
    ADNProvincia.Lista(d);
                ComboPro.DisplayMember = "Nombre";
                ComboPro.ValueMember = "IdProvincia";
            }
            else
            {
                Pais p = new Pais();
                p.IdPais =
    Convert.ToInt32(ComboDepa.SelectedValue);
                ComboPro.DataSource =
    ADNDepartamento.Lista(p);
                ComboPro.DisplayMember = "Nombre";
                ComboPro.ValueMember = "IdDepartamento";
            }
        }
    }

    void VerDatos() {
        dgvLista.Rows.Clear();
        Persona p = new Persona();
        p.ApePaterno = tbxFilApeP.Text.Trim();
        p.ApeMaterno = tbxFilApeM.Text.Trim();
        List<Feligres> lista = ADNFeligres.Lista(p,0);
        foreach (var fila in lista)
        {
            dgvLista.Rows.Add(fila.DatosToArray());
        }
    }
}

```

```

        }
        btnModificar.Enabled = false;
        btnEliminar.Enabled = false;
    }

    void GuardarDatos(Feligres fel)
    {
        DialogResult res = (idF == 0) ?
MessageBox.Show("¿Los datos ingresados son correctos?",  

                "Verificacion del Registro",  

                MessageBoxButtons.YesNo, MessageBoxIcon.Question)
                : DialogResult.Yes;
        if (res == DialogResult.Yes)
        {
            if (ADNPersona.Guardar(per))
            {
                int iddepapadre =
(chbxNohaypadre.Checked) ? 0 :
((chbxNacionalPadre.Checked)
? Convert.ToInt32(cbxDepaPadre.SelectedValue) :
Convert.ToInt32(cbxProPadre.SelectedValue));

                int idprofel = (chbxNacionalFel.Checked)
? Convert.ToInt32(cbxProFel.SelectedValue) : 0,
                idpropadre = (chbxNohaypadre.Checked) ?
0 :
((chbxNacionalPadre.Checked)
? Convert.ToInt32(cbxProPadre.SelectedValue) : 0),
                idpromadre = (chbxNacionalMadre.Checked)
? Convert.ToInt32(cbxProMadre.SelectedValue) : 0;

                fel.NatPadreDepa = new
Departamento(iddepapadre);
                fel.LugNacPro = new Provincia(idprofel);
                fel.NatPadrePro = new
Provincia(idpropadre);
                fel.NatMadrePro = new
Provincia(idpromadre);

                fel.DistritoFel =
(tbxDistritoFel.Visible) ? tbxDistritoFel.Text.Trim() : "";
                fel.DistritoPadre =
(tbxDistritoPadre.Visible) ? tbxDistritoPadre.Text.Trim() :
";
                fel.DistritoMadre =
(tbxDistritoMadre.Visible) ? tbxDistritoMadre.Text.Trim() :
";

                fel.NomPersona = new
Persona((per.IdPersona == 0) ? ADNPersona.MaxId() :
per.IdPersona);

                if (ADNFeligres.Guardar(fel))
                {
                    string r = (idF == 0) ? " guardados "
 : " actualizados ";
                    MessageBox.Show("Datos" + r + " con
exito.", "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                    btnNuevo.PerformClick(); VerDatos();
                }
            }
        }
    }
}

```

```

                else
                {
                    MessageBox.Show("Ha ocurrido un
problema.", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
                }
            }
            else
            {
                MessageBox.Show("Ha ocurrido un
problema.", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            }
        }
    }

    private void frmFeligres_Load(object sender,
EventArgs e)
{
    ColocarPro = false;
    CargarDepartamento(cbxDepaFel);
    CargarDepartamento(cbxDepaPadre);
    CargarDepartamento(cbxDepaMadre);
    cbxSexo.SelectedIndex = 0;
    cbxDepaPadre.SelectedIndex = 1;
    cbxDepaFel.SelectedIndex = 1;
    cbxDepaMadre.SelectedIndex = 1;
    dtpFecNac.MaxDate = DateTime.Today;
    ColocarPro = true; cbxDepaMadre.SelectedIndex =
0;
    cbxDepaFel.SelectedIndex = 0;
    cbxDepaPadre.SelectedIndex = 0;
    per.IdPersona = 0; idF = 0;
}

private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}

private void btnNuevo_Click(object sender, EventArgs
e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        tabControl1.SelectedIndex = 1;
        btnModificar.Enabled = false;
        btnEliminar.Enabled = false;
        Metodos.Limpiar(groupBox1);
        Metodos.Habilitar(groupBox1, false);
        //ColocarPro = true;
    }
    else
    {
        btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
        Metodos.Limpiar(groupBox3);
        Metodos.Limpiar(groupBox4);
    }
}

```

```

        Metodos.Limpiar(groupBox5);
Metodos.Limpiar(groupBox6);
        chbxPersonal.Visible = true;
chbxPersonal.Checked = false;
        idF = 0; per.IdPersona = 0;
tabControl1.SelectedIndex = 0;
        Metodos.Habilitar(groupBox1, true);
chbxNohaypadre.Checked = false;
        //ColocarPro = false;
    }
}

private void btnGuardar_Click(object sender,
EventArgs e)
{
    if (tbxApeP.TextLength > 0 &&
tbxNombres.TextLength > 0 && tbxNombres.TextLength > 0 &&
tbxPadre.TextLength > 0 &&
tbxMadre.TextLength > 0)
    {
        per.ApePaterno =
tbxApeP.Text.Trim().ToUpper();
        per.ApeMaterno =
tbxApeM.Text.Trim().ToUpper();
        per.Nombres =
tbxNombres.Text.Trim().ToUpper();
        per.FecNac = dtpFecNac.Value.Date;
        per.Sexo = cbxSexo.SelectedItem.ToString();

        int iddepafel = (chbxNacionalFel.Checked) ?
Convert.ToInt32(cbxDepaFel.SelectedValue) :

Convert.ToInt32(cbxProFel.SelectedValue),
        iddepamadre =
(chbxNacionalMadre.Checked) ?
Convert.ToInt32(cbxDepaMadre.SelectedValue) :

Convert.ToInt32(cbxProMadre.SelectedValue);

        Feligres fel = new Feligres();
        fel.IdFeligres = idF;
        fel.LugNacDepa = new
Departamento(iddepafel);
        fel.Padre =
Metodos.LetraTipoOracion(tbxPadre.Text.Trim());
        fel.Madre =
Metodos.LetraTipoOracion(tbxMadre.Text.Trim());
        fel.NatMadreDepa = new
Departamento(iddepamadre);

        if (ADNFeligres.ExisteAllFeligres(fel,per))
    {
        DialogResult res =
MessageBox.Show("Parece que el feligres ya ha sido
registrado.\n¿Desea Continuar?",

"Aviso", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
        if (res == DialogResult.Yes)
    {
        GuardarDatos(fel);
    }
}
}

```

```

        else if (ADNPersona.ExisteNomApe(per))
        {
            DialogResult res= MessageBox.Show("Se
han encontrado coincidencias de nombres y/o
apellidos.\n¿Desea Continuar?",
                                         "Aviso", MessageBoxButtons.YesNo,
                                         MessageBoxIcon.Question);
            if (res==DialogResult.Yes)
            {
                GuardarDatos(fel);
            }
            else
            {
                GuardarDatos(fel);
            }
        }
        else
        {
            MessageBox.Show("Faltan llenar datos.",
                            "Faltan Datos", MessageBoxButtons.OK, MessageBoxIcon.Stop);
        }
    }

    private void btnModificar_Click(object sender,
EventArgs e)
{
    var f = dgvLista.CurrentRow;
    idF = Convert.ToInt32(f.Cells[0].Value);
    per.IdPersona =
Convert.ToInt32(f.Cells[1].Value);
    tbxApeP.Text = f.Cells[2].Value.ToString();
    tbxApeM.Text = f.Cells[3].Value.ToString();
    tbxNombres.Text = f.Cells[4].Value.ToString();
    dtpFecNac.Value =
Convert.ToDateTime(f.Cells[5].Value);
    cbxSexo.SelectedItem =
f.Cells[6].Value.ToString();

    tbxMadre.Text = f.Cells[17].Value.ToString();
    chbxNacionalFel.Checked =
(Convert.ToInt32(f.Cells[7].Value) == 27) ? true : false;
    cbxDepaFel.Text = f.Cells[9].Value.ToString();
    cbxProFel.Text = f.Cells[8].Value.ToString();
    tbxDistritoFel.Text =
f.Cells[10].Value.ToString();

    if (f.Cells[12].Value.ToString() == "-----"
-----)
    {
        chbxNohaypadre.Checked = true;
    }
    else
    {
        tbxPadre.Text =
f.Cells[12].Value.ToString();
        chbxNacionalPadre.Checked =
(Convert.ToInt32(f.Cells[13].Value) == 27) ? true : false;
        cbxDepaPadre.Text =
f.Cells[15].Value.ToString();
        cbxProPadre.Text =
f.Cells[14].Value.ToString();
    }
}

```

```

        tbxDistritoPadre.Text =
f.Cells[16].Value.ToString();
    }

        chbxNacionalMadre.Checked =
(Convert.ToInt32(f.Cells[18].Value) == 27) ? true : false;
        cbxDepaMadre.Text =
f.Cells[20].Value.ToString();
        cbxProMadre.Text = f.Cells[19].Value.ToString();
        tbxDistritoMadre.Text =
f.Cells[21].Value.ToString();
        btnNuevo.PerformClick();
        chbxPersonal.Visible = false;
    }

    private void btnEliminar_Click(object sender,
EventArgs e)
{
    string apep =
(string)dgvLista.CurrentRow.Cells[2].Value,
apem =
(string)dgvLista.CurrentRow.Cells[3].Value,
nom =
(string)dgvLista.CurrentRow.Cells[4].Value;
    DialogResult res = MessageBox.Show("¿Realmente
desea eliminar a " + apep + " " + apem + ", " +
nom + "?", "Eliminar Registro",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (res == DialogResult.Yes)
    {
        Feligres fel = new Feligres();
        fel.IdFeligres =
Convert.ToInt32(dgvLista.CurrentRow.Cells[0].Value);
        fel.NomPersona = new
Persona(Convert.ToInt32(dgvLista.CurrentRow.Cells[1].Value));
;
        if (ADNFeligres.Eliminar(fel))
        {
            MessageBox.Show("Registro eliminado con
exito.", "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            VerDatos(); per.IdPersona = 0;
        }
        else
        {
            MessageBox.Show("No se puede
eliminar.\nRevise que el feligres no tenga sacramentos
asignados",
"Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
    }
}

private void btnCerrar_Click(object sender,
EventArgs e)
{
    this.Close();
}

private void chbxNacionalFel_CheckedChanged(object
sender, EventArgs e)

```

```

        {
            ColocarPro = false;
            if (chbxNacionalFel.Checked)
            {
                lblDepaFel.Text = "Departamento:";
                lblDepaFel.Location = new System.Drawing.Point(7, 23);
                CargarDepartamento(cbxDepaFel);
                lblProFel.Text = "Provincia:";
                lblProFel.Location = new System.Drawing.Point(30, 51);
                this.chbxNacionalFel.Location = new
                System.Drawing.Point(267, 78);
                lblDisFel.Visible = true;
                tbxDistritoFel.Visible = true;
            }
            else
            {
                lblDepaFel.Text = "Pais:";
                lblDepaFel.Location = new System.Drawing.Point(51, 23);
                CargarPais(cbxDepaFel);
                lblProFel.Text = "Ciudad:";
                lblProFel.Location = new System.Drawing.Point(38, 51);
                this.chbxNacionalFel.Location = new
                System.Drawing.Point(174, 78);
                lblDisFel.Visible = false;
                tbxDistritoFel.Visible = false;
            }
            cbxDepaFel.SelectedIndex = 1; ColocarPro = true;
            cbxDepaFel.SelectedIndex = 0;
        }

        private void chbxNacionalPadre_CheckedChanged(object
sender, EventArgs e)
{
    ColocarPro = false;
    if (chbxNacionalPadre.Checked)
    {
        lblDepaPadre.Text = "Departamento:";
        lblDepaPadre.Location = new System.Drawing.Point(12, 72);
        CargarDepartamento(cbxDepaPadre);
        lblProPadre.Text = "Provincia:";
        lblProPadre.Location = new System.Drawing.Point(35, 100);
        this.chbxNacionalPadre.Location = new
        System.Drawing.Point(271, 128);
        lblDisPadre.Visible = true;
        tbxDistritoPadre.Visible = true;
    }
    else
    {
        lblDepaPadre.Text = "Pais:";
        lblDepaPadre.Location = new System.Drawing.Point(59, 72);
        CargarPais(cbxDepaPadre);
        lblProPadre.Text = "Ciudad:";
        lblProPadre.Location = new System.Drawing.Point(46, 100);
        this.chbxNacionalPadre.Location = new
        System.Drawing.Point(153, 128);
        this.chbxNacionalPadre.Location = new
        System.Drawing.Point(171, 128);
        lblDisPadre.Visible = false;
        tbxDistritoPadre.Visible = false;
    }
    cbxDepaPadre.SelectedIndex = 1; ColocarPro =
    true; cbxDepaPadre.SelectedIndex = 0;
}

```

```

        }

        private void chbxNacionalMadre_CheckedChanged(object
sender, EventArgs e)
{
    ColocarPro = false;
    if (chbxNacionalMadre.Checked)
    {
        lblDepaMadre.Text = "Departamento:";
        lblDepaMadre.Location = new System.Drawing.Point(12, 71);
        CargarDepartamento(cbxDepaMadre);
        lblProMadre.Text = "Provincia:";
        lblProMadre.Location = new System.Drawing.Point(35, 99);
        this.chbxNacionalMadre.Location = new
System.Drawing.Point(271, 126);
        lblDisMadre.Visible = true;
        tbxDistritoMadre.Visible = true;
    }
    else
    {
        lblDepaMadre.Text = "Pais:";
        lblDepaMadre.Location = new System.Drawing.Point(59, 71);
        CargarPais(cbxDepaMadre);
        lblProMadre.Text = "Ciudad:";
        lblProMadre.Location = new System.Drawing.Point(46, 99);
        this.chbxNacionalMadre.Location = new
System.Drawing.Point(153, 130);
        this.chbxNacionalMadre.Location = new
System.Drawing.Point(171, 126);
        lblDisMadre.Visible = false;
        tbxDistritoMadre.Visible = false;
    }
    cbxDepaMadre.SelectedIndex = 1; ColocarPro =
true; cbxDepaMadre.SelectedIndex = 0;
}

private void cbxDepaFel_SelectedIndexChanged(object
sender, EventArgs e)
{
    CargarProvincia(chbxNacionalFel, cbxProFel,
cbxDepaFel);
}

private void
cbxDepaPadre_SelectedIndexChanged(object sender, EventArgs
e)
{
    CargarProvincia(chbxNacionalPadre, cbxProPadre,
cbxDepaPadre);
}

private void
cbxDepaMadre_SelectedIndexChanged(object sender, EventArgs
e)
{
    CargarProvincia(chbxNacionalMadre, cbxProMadre,
cbxDepaMadre);
}

private void tabControl1_SelectedIndexChanged(object
sender, EventArgs e)
{
}

```

```

        if (btnNuevo.Text == "Nuevo")
        {
            tabControl1.SelectedIndex = 0;
        }
        else
        {
            tabControl1.SelectedIndex = 1;
        }
    }

    private void dgvLista_Click(object sender, EventArgs e)
    {
        if (dgvLista.Rows.Count > 0)
        {
            btnModificar.Enabled = true;
            btnEliminar.Enabled = true;
        }
        else
        {
            btnModificar.Enabled = false;
            btnEliminar.Enabled = false;
        }
    }

    private void chbxPersonal_CheckedChanged(object sender, EventArgs e)
    {
        if (chbxPersonal.Checked)
        {
            btnBuscarPersonal.Visible = true;
            Metodos.Habilitar(groupBox3, false);
            btnBuscarPersonal.Enabled = true;
        }
        else
        {
            btnBuscarPersonal.Visible = false;
            Metodos.Habilitar(groupBox3, true);
        }
        Metodos.Limpiar(groupBox3); per.IdPersona = 0;
    }

    private void ManejadorLetras(object sender, KeyPressEventArgs e)
    {
        Metodos.SoloLetras(e);
    }

    private void btnBuscarPersonal_Click(object sender, EventArgs e)
    {
        per.IdPersona = 0; invoca = true;
        frmBuscarPersonal f = new frmBuscarPersonal();
        f.ShowDialog();
        if (per.IdPersona > 0)
        {
            tbxApeP.Text = per.ApePaterno; tbxApeM.Text
            = per.ApeMaterno; tbxNombres.Text = per.Nombres; cbxSexo.Text
            = per.Sexo; dtpFecNac.Value = per.FecNac;
        }
    }

```

```

        else
        { per.IdPersona = 0; Metodos.Limpiar(groupBox3);
    }
    invoca = false;
}

private void tbxPadre_KeyPress(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
}

private void ManejadorFiltro(object sender,
KeyPressEventArgs e)
{
    if (e.KeyChar == (char)Keys.Enter)
    {
        btnBuscar.PerformClick();
    }
}

private void chbxNohaypadre_CheckedChanged(object
sender, EventArgs e)
{
    if (chbxNohaypadre.Checked)
    {
        tbxPadre.Text = "-----";
        tbxPadre.Enabled = false;
        cbxDepaPadre.Enabled = false;
        cbxProPadre.Enabled = false;
        tbxDistritoPadre.Enabled = false;
        chbxNacionalPadre.Enabled = false;
        tbxDistritoPadre.Clear();
    }
    else
    {
        tbxPadre.Clear(); chbxNacionalPadre.Enabled
= true;
        tbxPadre.Enabled = true;
        cbxDepaPadre.Enabled = true;
        cbxProPadre.Enabled = true;
        tbxDistritoPadre.Enabled = true;
    }
}
}
}

```

- frnLibroSacramento.cs

```

using Entidades;
using AccesoDatosNegocio;
using Utilidades;

namespace Presentacion
{
    public partial class frmLibroSacramento :
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmLibroSacramento()
        {

```

```

        InitializeComponent();
    }

    static int idL = 0;
    static bool estb = false, estc = false, estm =
false;

    void VerDatos() {
        dgvLista.Rows.Clear();
        string tipo = (rbnBautismo.Checked) ? "Bautismo"
:
        ((rbnConfirmacion.Checked) ? "Confirmacion"
: "Matrimonio");
        List<LibroSacramento> lista =
ADNLibroSacramento.Lista(new LibroSacramento(tipo));
        foreach (var ele in lista)
        {
            dgvLista.Rows.Add(ele.DatosToArray());
        }
        btnCerrarLibro.Enabled = false;
        btnModificar.Enabled = false;
    }

    private void frmLibroSacramento_Load(object sender,
EventArgs e)
{
    this.ClientSize = new System.Drawing.Size(420,
315);
    estb = frmPrincipal.b; estc =frmPrincipal.c;
    estm = frmPrincipal.m;
    if (estb==false)
    {
        rbnBautismo.Checked = true;
        cbxTipo.Items.Add("Bautismo");
    }
    if (estc == false)
    {
        rbnConfirmacion.Checked = (estb == false) ?
false : true;
        cbxTipo.Items.Add("Confirmacion");
    }
    if (estm == false)
    {
        rbnMatrimonio.Checked = (estb == false) ?
false : ((estc == false) ? false : true);
        cbxTipo.Items.Add("Matrimonio");
    }
    rbnBautismo.Enabled = !estb;
    rbnConfirmacion.Enabled = !estc; rbnMatrimonio.Enabled =
!estm;
    VerDatos(); cbxTipo.SelectedIndex = 0;
}

private void ManejadorNumeros(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloNumeros(e);
}

private void btnNuevo_Click(object sender, EventArgs
e)
{

```

```

        if (btnNuevo.Text == "Nuevo")
        {
            btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
            this.ClientSize = new
System.Drawing.Size(420, 500);
            btnModificar.Enabled = false;
            btnCerrarLibro.Enabled = false; dgvLista.Enabled = false;
            tbxFechaApertura.Text =
DateTime.Now.ToShortDateString();
            tbxEstado.Text = "Abierto";
            tbxRegistros.Text = "0";
            Metodos.Habilitar(groupBox1, false);
        }
        else
        {
            btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
            this.ClientSize = new
System.Drawing.Size(420, 315);
            Metodos.Limpiar(groupBox3); cbxTipo.Enabled
= true;
            dgvLista.Enabled = true; idL = 0;
            rbnBautismo.Enabled = !estb;
            rbnConfirmacion.Enabled = !estc; rbnMatrimonio.Enabled =
!estm;
        }
    }

    private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}

private void btnSalir_Click(object sender, EventArgs
e)
{
    this.Close();
}

private void cbxTipo_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (cbxTipo.SelectedIndex==1)
    {
        nudNumPorFolio.Value = new decimal(new int[]
{ 38, 0, 0, 0 });
    }
    else
    {
        nudNumPorFolio.Value = new decimal(new int[]
{ 1, 0, 0, 0 });
    }
}

private void btnGuardar_Click(object sender,
EventArgs e)
{
    LibroSacramento l = new LibroSacramento();
    l.IdLibro = idL;
    l.Tipo = cbxTipo.SelectedItem.ToString();
}

```

```

        l.Numero = Convert.ToInt32(nudNumero.Value);
        l.MaximoFolios = Convert.ToInt32(nudMax.Value);
        l.NumeroPorFolio =
Convert.ToInt32(nudNumPorFolio.Value);
        if (ADNLibroSacramento.ExisteLibro(l))
        {
            MessageBox.Show("El libro ya existe",
"Duplicado", MessageBoxButtons.OK, MessageBoxIcon.Stop);
        }
        else
        {

            DialogResult res = (idL==0) ?
MessageBox.Show("¿El tipo del libro es correcto?\nEste dato
no se podra modificar mas adelante.",
"Verificar Datos",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) :
DialogResult.Yes;
            if (res==DialogResult.Yes)
            {
                if (ADNLibroSacramento.Guardar(l))
                {
                    string r = (idL == 0) ? " guardados
" : " actualizados ";
                    string r1 = (idL==0) ? "\nNOTA:Una
vez ingresado registros en el libro, no podra
modificarlo.:";;
                    MessageBox.Show("Datos" + r + "con
exito." + r1, "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                    btnNuevo.PerformClick(); VerDatos();
                }
                else
                {
                    MessageBox.Show("El libro ya
existe", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
                }
            }
        }
    }

    private void frmLibroSacramento_FormClosing(object
sender, FormClosingEventArgs e)
{
    idL = 0;
}

private void btnModificar_Click(object sender,
EventArgs e)
{
    DataGridViewRow f = dgvLista.CurrentRow;
    btnNuevo.PerformClick();
    idL = Convert.ToInt32(f.Cells[0].Value);
    cbxTipo.Text = f.Cells[1].Value.ToString();
    nudNumero.Value = new decimal(new int[]
{Convert.ToInt32(f.Cells[2].Value),0,0,0});
    nudMax.Value = new decimal(new
int[]{Convert.ToInt32(f.Cells[3].Value),0,0,0});
    nudNumPorFolio.Value = new decimal(new
int[]{Convert.ToInt32(f.Cells[4].Value),0,0,0});
    tbxRegistros.Text = f.Cells[5].Value.ToString();
}

```

```

        tbxFechaApertura.Text =
f.Cells[6].Value.ToString();
        tbxFechaCierre.Text =
f.Cells[7].Value.ToString();
        tbxEstado.Text = f.Cells[8].Value.ToString();
        cbxTipo.Enabled = false;
    }

private void dgvLista_Click(object sender, EventArgs
e)
{
    if (dgvLista.Rows.Count > 0)
    {
        DataGridViewRow f = dgvLista.CurrentRow;
        int n3 =
Convert.ToInt32(f.Cells[3].Value.ToString()),
n4 =
Convert.ToInt32(f.Cells[4].Value.ToString()),
n5 =
Convert.ToInt32(f.Cells[5].Value.ToString());

        if (f.Cells[8].Value.ToString() ==
"Abierto")
        {
            btnCerrarLibro.Image =
global::Presentacion.Resource1.Moleskine_Blanck_Book_icon;
            btnCerrarLibro.Text = "Cerrar Libro";
            btnCerrarLibro.Enabled = true;
            btnModificar.Enabled =
=(Convert.ToInt32(f.Cells[5].Value)>0) ? false : true;
        }
        else
        {
            btnCerrarLibro.Image =
global::Presentacion.Resource1.book_open_icon;
            btnCerrarLibro.Text = "Reaperturar";
            btnModificar.Enabled = false;
            btnCerrarLibro.Enabled = (n3*n4==n5) ?
false : true;
        }
    }
}

private void btnCerrarLibro_Click(object sender,
EventArgs e)
{
    DataGridViewRow f = dgvLista.CurrentRow;
    string n = f.Cells[2].Value.ToString(), s =
f.Cells[1].Value.ToString();
    string es = (btnCerrarLibro.Text.Equals("Cerrar
Libro")) ? "cerrar" : "reaperturar";
    DialogResult res = MessageBox.Show("¿Está
seguro(a) que desea "+es+" el libro "+n+" de "+s+"?",
btnCerrarLibro.Text+" Libro",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (res==DialogResult.Yes)
    {
        LibroSacramento l = new LibroSacramento();

l.IdLibro=Convert.ToInt32(f.Cells[0].Value.ToString());
        l.Abierto=(es.Equals("cerrar")) ? false :
true;
    }
}

```

```

        if (ADNLibroSacramento.CambiarEstado(1))
        {
            string me = (es.Equals("cerrar")) ?
"cerrado" : "reaperturado";
            MessageBox.Show("Libro "+me+" con
éxito", "Exito",
                MessageBoxButtons.OK,
                MessageBoxIcon.Information);
            VerDatos();
        }
        else
        {
            MessageBox.Show("Ha ocurrido un
problema", "Error",
                MessageBoxButtons.OK,
                MessageBoxIcon.Error);
        }
    }

    private void rbnBautismo_CheckedChanged(object
sender, EventArgs e)
{
    VerDatos();
}

private void rbnConfirmacion_CheckedChanged(object
sender, EventArgs e)
{
    VerDatos();
}

private void rbnMatrimonio_CheckedChanged(object
sender, EventArgs e)
{
    VerDatos();
}

}
}

```

- frmMatrimonio.cs

```

using Entidades;
using AccesoDatosNegocio;
using Utilidades;

namespace Presentacion
{
    public partial class frmMatrimonio :
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmMatrimonio()
        {
            InitializeComponent();
        }

        static int idM = 0;
        static Matrimonio matrimonio = null;
        static LibroSacramento libro = null;
        static bool mod = false;
    }
}

```

```

        static int edado = 0, edada=0;
        static DateTime fecbau1 = DateTime.Today;

        public static Feligres esposo = new Feligres();
        public static Feligres esposa = new Feligres();
        public static Templo templo = new Templo();
        public static Templo parroesposo = new Templo();
        public static Templo parroesposa = new Templo();
        public static bool esp1 = false;
        public static bool esp2 = false;
        public static bool invoca = false;

        void VerDatos()
        {
            dgvLista.Rows.Clear();
            Persona p1 = new Persona();
            p1.ApePaterno = tbxFilApeP1.Text.Trim();
            p1.ApeMaterno = tbxFilApeM1.Text.Trim();
            Persona p2 = new Persona();
            p2.ApePaterno = tbxFilApeP2.Text.Trim();
            p2.ApeMaterno = tbxFilApeM2.Text.Trim();
            List<Matrimonio> lista = ADNMatrimonio.Lista(p1,
p2);
            foreach (var ele in lista)
            {
                dgvLista.Rows.Add(ele.DatosToArray());
            }
        }

        void VerLibros(int num)
        {
            cbxLibro.ValueMember = "IdLibro";
            cbxLibro.DataSource = (num > 0) ?
ADNLibroSacramento.Lista2(new LibroSacramento(num))
                : ADNLibroSacramento.Lista2(new
LibroSacramento("Matrimonio"));
            cbxLibro.DisplayMember = "Numero";
        }

        void CalcularFolioNumero()
        {
            int num = 0, folio = 0;
            LibroSacramento lib = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
            matrimonio = ADNMatrimonio.FolioNumero(lib);
            if (matrimonio == null || chbxContinuar.Checked
== false)
            {
                num = 1; folio = 1;
            }
            else
            {
                num = matrimonio.Numero + 1;
                folio = (matrimonio.Numero %
libro.NumeroPorFolio == 0) ?
                    matrimonio.Folio + 1 : matrimonio.Folio;
            }
            nudFolio.Value = new decimal(new int[] { folio,
0, 0, 0 });
            nudNumero.Value = new decimal(new int[] { num,
0, 0, 0 });
            CalcularMaxMinNumero();
        }
    }
}

```

```

        void CalcularMaxMinNumero()
        {
            int fol = Convert.ToInt32(nudFolio.Value);
            nudNumero.Minimum = new decimal(new int[] {
                ((fol - 1) * libro.NumeroPorFolio) + 1, 0, 0, 0 });
            nudNumero.Maximum = new decimal(new int[] { fol
                * libro.NumeroPorFolio, 0, 0, 0 });
        }

        void HabilitarConMod(bool est)
        {
            cbxLibro.Enabled = est; nudFolio.Enabled = est;
            nudNumero.Enabled = est; chbxContinuar.Enabled =
            est;
            btnBuscarEsposo.Enabled = est;
            btnBuscarEsposa.Enabled = est;
            tbxEstCivilEsposo.Enabled = !est;
            tbxEstCivilEsposa.Enabled = !est;
        }

        private void frmMatrimonio_Load(object sender,
        EventArgs e)
        {
            Metodos.CerrarForm<frmLibroSacramento>(); idM =
            0;
            VerLibros(0); mod = false;
            dtpFecha.MaxDate = DateTime.Today;
            dtpFechaBauEsposa.MaxDate = DateTime.Today;
            dtpFechaBauEsposo.MaxDate = DateTime.Today;
            esposo.IdFeligrés = 0; esposa.IdFeligrés = 0;
            templo.IdTemplo = 0;
            parroesposo.IdTemplo = 0; parroesposa.IdTemplo =
            0;
        }

        private void tabControl1_SelectedIndexChanged(object
        sender, EventArgs e)
        {
            if (btnNuevo.Text == "Nuevo")
            {
                tabControl1.SelectedIndex = 0;
            }
            else if (tabControl1.SelectedIndex==0)
            {
                tabControl1.SelectedIndex = 1;
            }
        }

        private void btnBuscar_Click(object sender,
        EventArgs e)
        {
            VerDatos();
        }

        private void btnCerrar_Click(object sender,
        EventArgs e)
        {
            this.Close();
        }
    
```

```

        private void btnNuevo_Click(object sender, EventArgs e)
    {
        if (btnNuevo.Text == "Nuevo")
        {
            btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
            tabControl1.SelectedIndex = 1;
            btnModificar.Enabled = false;
            Metodos.Limpiar(groupBox1);
            Metodos.Habilitar(groupBox1, false);
        }
        else
        {
            btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
            Metodos.Limpiar(groupBox3);
            Metodos.Limpiar(groupBox4);
            Metodos.Limpiar(groupBox5);
            Metodos.Limpiar(groupBox6);
            Metodos.Limpiar(groupBox7);
            idM = 0; esposo.IdFeligrés = 0;
            esposa.IdFeligrés = 0; templo.IdTemplo = 0;
            if (btnBuscarParroBauEsposo.Enabled)
            {
                Metodos.Habilitar(groupBox5, false);
            }
            if (btnBuscarParroBauEsposa.Enabled)
            {
                Metodos.Habilitar(groupBox6, false);
            }
            tbxEstCivilEsposo.Enabled = false;
            tbxEstCivilEsposa.Enabled = false;
            Metodos.Habilitar(groupBox1, true);
            tabControl1.SelectedIndex = 0;
        }
    }

    private void cbxLibro_SelectedIndexChanged(object sender, EventArgs e)
    {
        LibroSacramento lib = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
        libro = ADNLibroSacramento.DatosLibro(lib);
        nudFolio.Maximum = new decimal(new int[] {
libro.MaximoFolios, 0, 0, 0 });
        if (mod == false)
        {
            CalcularFolioNúmero();
        }
    }

    private void nudFolio_ValueChanged(object sender, EventArgs e)
    {
        CalcularMaxMinNúmero();
    }

    private void chbxContinuar_CheckedChanged(object sender, EventArgs e)
    {
        CalcularFolioNúmero();
    }

```

```

        }

        private void btnBuscarEsposo_Click(object sender,
EventArgs e)
{
    esposo.IdFeligres = 0; esp1 = true;
    frmBuscarFeligres f = new frmBuscarFeligres();
f.ShowDialog();
    if (esposo.IdFeligres>0)
    {
        tbxEsposo.Text =
esposo.NomPersona.ApePaterno + " " +
esposo.NomPersona.ApeMaterno + ", " +
esposo.NomPersona.Nombres;
        tbxPadreEsposo.Text = esposo.Padre;
        tbxMadreEsposo.Text = esposo.Madre;
        tbxLugFecNacEsposo.Text =
esposo.LugNacDepa.Nombre + " - " + esposo.LugNacPro.Nombre +
", El día " +
esposo.NomPersona.FecNac.ToShortDateString();
        edado =
Metodos.CalcularEdad(esposo.NomPersona.FecNac);
        tbxEdadEsposo.Text = edado + " años";
        Bautismo bau = ADNBautismo.Datos(esposo,0);
        if (bau == null)
        {
            Metodos.Limpiar(groupBox5);
            dtpFechaBauEsposo.Enabled = true;
        btnBuscarParroBauEsposo.Enabled = true;
            DateTime fecnac =
esposo.NomPersona.FecNac;
            dtpFechaBauEsposo.MinDate = new
System.DateTime(fecnac.Year, fecnac.Month, fecnac.Day, 0, 0,
0, 0);
        }
        else
        {
            Metodos.Habilitar(groupBox5, false);
            parroesposo = bau.NomTemplo;
            tbxParroBauEsposo.Text =
parroesposo.Nombre + " de " + parroesposo.Distrito + " "
Pisco";
            dtpFechaBauEsposo.MinDate = new
System.DateTime(1753, 1, 1, 0, 0, 0, 0);
            dtpFechaBauEsposo.Value = bau.Fecha;
        }
        tbxEstCivilEsposo.Clear();
        tbxEstCivilEsposo.Enabled = true;
    }
    else
    {
        Metodos.Limpiar(groupBox3);
        Metodos.Limpiar(groupBox5);
        tbxEstCivilEsposo.Enabled = false;
    }
    esp1 = false;
}

private void btnBuscarEsposa_Click(object sender,
EventArgs e)
{
    esposo.IdFeligres = 0; esp2 = true;
}

```

```

        frmBuscarFeligres f = new frmBuscarFeligres();
f.ShowDialog();
        if (esposa.IdFeligres > 0)
        {
            tbxEsposa.Text =
esposa.NomPersona.ApePaterno + " " +
esposa.NomPersona.ApeMaterno + ", " +
                espresa.NomPersona.Nombres;
            tbxPadreEsposa.Text = sposa.Padre;
            tbxMadreEsposa.Text = sposa.Madre;
            tbxLugFecNacEsposa.Text =
esposa.LugNacDepa.Nombre + " - " + sposa.LugNacPro.Nombre +
", El día " +
esposa.NomPersona.FecNac.ToShortDateString();
            edada =
Metodos.CalcularEdad(esposa.NomPersona.FecNac);
            tbxEdadEsposa.Text = edada + " años";
            Bautismo bau = ADNBautismo.Datos(esposa,0);
            if (bau == null)
            {
                Metodos.Limpiar(groupBox6);
                dtpFechaBauEsposa.Enabled = true;
            btnBuscarParroBauEsposa.Enabled = true;
                DateTime fecnac =
esposa.NomPersona.FecNac;
                dtpFechaBauEsposa.MinDate = new
System.DateTime(fecnac.Year, fecnac.Month, fecnac.Day, 0, 0,
0, 0);
            }
            else
            {
                Metodos.Habilitar(groupBox6, false);
                parroesposa = bau.NomTemplo;
                tbxParroBauEsposa.Text =
parroesposa.Nombre + " de " + parroesposa.Distrito + "
Pisco";
                dtpFechaBauEsposa.MinDate = new
System.DateTime(1753, 1, 1, 0, 0, 0, 0);
                dtpFechaBauEsposa.Value = bau.Fecha;
            }
            tbxEstCivilEsposa.Clear();
            tbxEstCivilEsposa.Enabled = true;
        }
        else
        {
            Metodos.Limpiar(groupBox4);
            Metodos.Limpiar(groupBox6);
            tbxEstCivilEsposa.Enabled = false;
        }
        esp2 = false;
    }

    private void btnBuscarParroBauEsposo_Click(object
sender, EventArgs e)
    {
        parroesposo.IdTemplo = 0; esp1 = true;
        frmBuscarTemplo f = new frmBuscarTemplo();
f.ShowDialog();
        if (parroesposo.IdTemplo>0)
        {
            tbxParroBauEsposa.Text = parroesposo.Nombre
+ " de " + parroesposo.Tipo;

```

```

        }
        else
        {
            tbxParroBauEsposo.Clear();
        }
        esp1 = false;
    }

    private void btnBuscarParroBauEsposa_Click(object sender, EventArgs e)
    {
        parroesposa.IdTemplo = 0; esp2 = true;
        frmBuscarTemplo f = new frmBuscarTemplo();
        f.ShowDialog();
        if (parroesposa.IdTemplo > 0)
        {
            tbxParroBauEsposa.Text = parroesposa.Nombre
+ " de " + parroesposa.Tipo;
        }
        else
        {
            tbxParroBauEsposa.Clear();
        }
        esp2 = false;
    }

    private void btnBuscarTemplo_Click(object sender, EventArgs e)
    {
        templo.IdTemplo = 0; invoca = true;
        frmBuscarTemplo f = new frmBuscarTemplo();
        f.ShowDialog();
        if (templo.IdTemplo > 0)
        {
            tbxTemplo.Text = templo.Nombre + " de " +
templo.Tipo;
        }
        else
        {
            tbxTemplo.Clear();
        }
        invoca = false;
    }

    private void btnGuardar_Click(object sender, EventArgs e)
    {
        if (esposo.IdFeligres>0 && esposa.IdFeligres>0
&& parroesposo.IdTemplo>0 && parroesposa.IdTemplo>0 &&
templo.IdTemplo>0 &&
tbxEstCivilEsposo.TextLength>0 &&
tbxEstCivilEsposa.TextLength>0 &&
tbxPadrino.TextLength>0 &&
tbxMadrina.TextLength>0 && tbxTestigo1.TextLength>0 &&
tbxTestigo2.TextLength>0 &&
tbxMinistro.TextLength>0)
        {
            Matrimonio matri = new Matrimonio();
            matri.IdMatrimonio = idM;
            matri.Libro = new
LibroSacramento(Convert.ToInt32(cbxLibro.SelectedValue));
        }
    }
}

```

```

matri.Folio =
Convert.ToInt32(nudFolio.Value);
matri.Numero =
Convert.ToInt32(nudNumero.Value);
if (ADNMatrimonio.ExisteLFN(matri))
{
    string nom =
ADNMatrimonio.FeligresLFN(matri);
    MessageBox.Show("Ya existe un registro
en el libro, folio y número.\nFeligres: " + nom, "Registro
Parroquial existente",
                MessageBoxButtons.OK,
                MessageBoxIcon.Stop);
}
else
{
    DialogResult res = (idM == 0) ?
MessageBox.Show("¿El libro, folio y número son
correctos?.\n" +
                    "Estos valores no se podran
modificar mas adelante", "Verificacion del Registro
Parroquial",
                MessageBoxButtons.YesNo,
                MessageBoxIcon.Question) : DialogResult.Yes;
    if (res == DialogResult.Yes)
    {
        matri.Esposo = esposo;
        matri.EstCivilEsposo =
tbxEstCivilEsposo.Text.Trim();
        matri.EdadEsposo = edado;
        matri.TemBauEsposo = parroesposo;
        matri.FecBauEsposo =
dtpFechaBauEsposo.Value;
        matri.Esposa = esposa;
        matri.EstCivilEsposa =
tbxEstCivilEsposa.Text.Trim();
        matri.EdadEsposa = edada;
        matri.TemBauEsposa = parroesposa;
        matri.FecBauEsposa =
dtpFechaBauEsposa.Value;
        matri.NomTemplo = templo;
        matri.Fecha = dtpFecha.Value;
        matri.Ministro =
tbxMinistro.Text.Trim();
        matri.Padrino =
tbxPadrino.Text.Trim();
        matri.Madrina =
tbxMadrina.Text.Trim();
        matri.Testigo1 =
tbxTestigo1.Text.Trim();
        matri.Testigo2 =
tbxTestigo2.Text.Trim();
        matri.Anotaciones =
tbxNota.Text.Trim();
        if (ADNMatrimonio.Guardar(matri))
        {
            string r = (idM == 0) ?
"guardados" : "actualizados";
            MessageBox.Show("Datos " + r +
"con exito.", "Exito", MessageBoxButtons.OK,
                MessageBoxIcon.Information);
        }
    }
}

```

```

        btnNuevo.PerformClick();
    VerDatos();
}
else
{
    MessageBox.Show("Ha ocurrido un
    problema.", "Error", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
}
}
else
{
    MessageBox.Show("Faltan Datos", "Datos",
    MessageBoxButtons.OK, MessageBoxIcon.Stop);
}
}

private void btnModificar_Click(object sender,
EventArgs e)
{
    mod = true;
    DataGridViewRow f = dgvLista.CurrentRow;
    idM = Convert.ToInt32(f.Cells[0].Value);
    esposo.IdFeligrés =
    Convert.ToInt32(f.Cells[1].Value);
    tbxEsposo.Text = f.Cells[2].Value.ToString();
    tbxPadreEsposo.Text =
    f.Cells[3].Value.ToString();
    tbxMadreEsposo.Text =
    f.Cells[4].Value.ToString();
    tbxEstCivilEsposo.Text =
    f.Cells[5].Value.ToString();
    tbxEdadEsposo.Text = f.Cells[7].Value + " años";

    Feligres aux1 = ADNFeligres.LugarNac(esposo);
    tbxLugFecNacEsposo.Text = aux1.LugNacDepa.Nombre
    + " - " + aux1.LugNacPro.Nombre + ", El día "
    + Convert.ToDateTime(f.Cells[6].Value).ToShortDateString();

    parroesposo.IdTemplo =
    Convert.ToInt32(f.Cells[8].Value);
    if (parroesposo.IdTemplo > 1)
    {
        btnBuscarParroBauEsposo.Enabled = true;
        dtpFechaBauEsposo.Enabled = true;
    }
    Templo aux2 =
    ADNTemplo.NombreLugar(parroesposo);
    tbxParroBauEsposo.Text = aux2.Nombre + " de " +
    aux2.LugarPro.Nombre + ", " + aux2.LugarDepa.Nombre;
    dtpFechaBauEsposo.Value =
    Convert.ToDateTime(f.Cells[9].Value);

    esposa.IdFeligrés =
    Convert.ToInt32(f.Cells[10].Value);
    tbxEsposa.Text = f.Cells[11].Value.ToString();
    tbxPadreEsposa.Text =
    f.Cells[12].Value.ToString();
}

```

```

        tbxMadreEsposa.Text =
f.Cells[13].Value.ToString();
        tbxEstCivilEsposa.Text =
f.Cells[14].Value.ToString();
        tbxEDadEsposa.Text = f.Cells[16].Value + "
años";

        aux1 = ADNFeligres.LugarNac(esposa);
        tbxLugFecNacEsposa.Text = aux1.LugNacDepa.Nombre
+ " - " + aux1.LugNacPro.Nombre + ", El día " +
Convert.ToDateTime(f.Cells[15].Value).ToShortDateString();

        parroesposa.IdTemplo =
Convert.ToInt32(f.Cells[17].Value);
        if (parroesposa.IdTemplo > 1)
{
            btnBuscarParroBauEsposa.Enabled = true;
dtpFechaBauEsposa.Enabled = true;
}
        aux2 = ADNTemplo.NombreLugar(parroesposa);
        tbxParroBauEsposa.Text = aux2.Nombre + " de " +
aux2.LugarPro.Nombre + ", " + aux2.LugarDepa.Nombre;
        dtpFechaBauEsposa.Value =
Convert.ToDateTime(f.Cells[18].Value);

        templo.IdTemplo =
Convert.ToInt32(f.Cells[19].Value);
        tbxTemplo.Text = f.Cells[20].Value.ToString();
        dtpFecha.Value =
Convert.ToDateTime(f.Cells[21].Value);
        tbxMinistro.Text = f.Cells[22].Value.ToString();
        tbxPadrino.Text = f.Cells[23].Value.ToString();
        tbxMadrina.Text = f.Cells[24].Value.ToString();
        tbxTestigo1.Text = f.Cells[25].Value.ToString();
        tbxTestigo2.Text = f.Cells[26].Value.ToString();
        if (f.Cells[32].Value.ToString() == "False")
{

    VerLibros(Convert.ToInt32(f.Cells[27].Value));
}
        cbxLibro.SelectedValue =
Convert.ToInt32(f.Cells[27].Value);
        nudFolio.Value = new decimal(new int[] {
Convert.ToInt32(f.Cells[28].Value), 0, 0, 0 });
        nudNumero.Value = new decimal(new int[] {
Convert.ToInt32(f.Cells[29].Value), 0, 0, 0 });
        tbxNota.Text = f.Cells[31].Value.ToString();

        btnNuevo.PerformClick(); HabilitarConMod(false);
}

private void dgvLista_Click(object sender, EventArgs e)
{
    btnModificar.Enabled = (dgvLista.Rows.Count > 0)
? true : false;
}

private void dtpFechaBauEsposo_ValueChanged(object sender, EventArgs e)
{

```

```

        DateTime fec = (dtpFechaBauEsposo.Value >
dtpFechaBauEsposo.Value) ?
            dtpFechaBauEsposo.Value :
((dtpFechaBauEsposo.Value < dtpFechaBauEsposo.Value) ?
            dtpFechaBauEsposo.Value :
dtpFechaBauEsposo.Value);
            dtpFecha.MinDate = new System.DateTime(fec.Year,
fec.Month, fec.Day, 0, 0, 0);
        }

        private void dtpFechaBauEsposo_ValueChanged(object
sender, EventArgs e)
{
    DateTime fec = (dtpFechaBauEsposo.Value >
dtpFechaBauEsposo.Value) ?
            dtpFechaBauEsposo.Value :
((dtpFechaBauEsposo.Value < dtpFechaBauEsposo.Value) ?
            dtpFechaBauEsposo.Value :
dtpFechaBauEsposo.Value);
            dtpFecha.MinDate = new System.DateTime(fec.Year,
fec.Month, fec.Day, 0, 0, 0);
}

        private void frmMatrimonio_SizeChanged(object
sender, EventArgs e)
{
    if (btnNuevo.Text.Equals("Cancelar"))
    {
        btnNuevo.PerformClick();
    }
    VerDatos();
}

        private void ManejadorLetras(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
}

        private void tbxMinistro_KeyPress(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
    if (Char.IsPunctuation(e.KeyChar))
    { e.Handled = false; }
}

        private void ManejadorFiltro(object sender,
KeyPressEventArgs e)
{
    if (e.KeyChar == (char)Keys.Enter)
    {
        btnBuscar.PerformClick();
    }
}
}

```

- frmPersonal.cs  
`using Utilidades;`

```

using Entidades;
using AccesoDatosNegocio;

namespace Presentacion
{
    public partial class frmPersonal : 
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmPersonal()
        {
            InitializeComponent();
        }

        static int idP = 0;

        public static Persona per = new Persona();
        public static bool invoca = false;

        void VerDatos()
        {
            dgvLista.Rows.Clear();
            Persona p = new Persona();
            p.ApePaterno = tbxFilApeP.Text.Trim();
            p.ApeMaterno = tbxFilApeM.Text.Trim();
            List<Personal> lista = ADNPersonal.Lista(p,0);
            foreach (var fila in lista)
            {
                dgvLista.Rows.Add(fila.DatosToArray());
            }
            btnModificar.Enabled = false;
            btnEliminar.Enabled = false;
        }

        void VerTipos()
        {
            cbxTipo.DataSource = ADNTipos.Lista(new
Tipos("Personal"));
            cbxTipo.DisplayMember = "Nombre";
            cbxTipo.ValueMember = "IdTipo";
        }

        void GuardarDatos()
        {
            DialogResult res = (idP == 0) ?
MessageBox.Show("¿Los datos ingresados son correctos?", 
                "Verificacion del Registro",
                MessageBoxButtons.YesNo, MessageBoxIcon.Question)
                : DialogResult.Yes;
            if (res==DialogResult.Yes)
            {
                if (ADNPersonal.Guardar(per))
                {
                    Personal p = new Personal();
                    p.IdPersonal = idP;
                    p.NomPersona = new
                    Persona((per.IdPersonal == 0) ? ADNPersonal.MaxId() :
                    per.IdPersonal);
                    p.NomTipo = new
                    Tipos(Convert.ToInt32(cbxTipo.SelectedValue));
                    p.Domicilio = tbxDomicilio.Text.Trim();
                    p.NumCelular = mtbxCelular.Text;
                    p.NumTelefono = mtbxTelefono.Text;
                }
            }
        }
    }
}

```

```

                if (ADNPPersonal.Guardar(p))
                {
                    string r = (idP == 0) ? " guardados"
" : " actualizados ";
                    MessageBox.Show("Datos" + r + " con
exito.", "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                    btnNuevo.PerformClick(); VerDatos();
per.IdPersona = 0;
                }
                else
                {
                    MessageBox.Show("Ha ocurrido un
problema.", "Error", MessageBoxButtons.OK,
 MessageBoxIcon.Error);
                }
            }
            else
            {
                MessageBox.Show("Ha ocurrido un
problema.", "Error", MessageBoxButtons.OK,
 MessageBoxIcon.Error);
            }
        }
    }

    private void frmPersonal_Load(object sender,
EventArgs e)
{
    idP = 0; VerTipos(); cbxSexo.SelectedIndex = 0;
invoca = false;
    dtpFecNac.MaxDate = DateTime.Now;
}

    private void btnCerrar_Click(object sender,
EventArgs e)
{
    this.Close();
}

    private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}

    private void btnNuevo_Click(object sender, EventArgs
e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        tabControl1.SelectedIndex = 1;
        btnModificar.Enabled = false;
        btnEliminar.Enabled = false;
        Metodos.Limpiar(groupBox1);
        Metodos.Habilitar(groupBox1, false);
    }
    else
    {
}
}

```

```

        btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
        idP = 0; per.IdPersona = 0;
tabControl1.SelectedIndex = 0;
        Metodos.Habilitar(groupBox1, true);
        Metodos.Limpiar(groupBox2);
Metodos.Limpiar(groupBox3);
        chbxFeligres.Visible = true;
chbxFeligres.Checked = false;
    }
}

private void tabControl1_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        tabControl1.SelectedIndex = 0;
    }
    else
    {
        tabControl1.SelectedIndex = 1;
    }
}

private void btnGuardar_Click(object sender,
EventArgs e)
{
    int loncel = mtbxCelular.Text.Length, lontel =
mtbxTelefono.Text.Length;
    if (txxApeP.TextLength > 0 && txxApeM.TextLength
> 0 && tbxNombres.TextLength > 0)
    {
        if ((loncel==0 || loncel==9) && (lontel==0
|| lontel==9))
        {
            per.ApePaterno =
Metodos.LetraTipoOracion(txxApeP.Text.Trim());
            per.ApeMaterno =
Metodos.LetraTipoOracion(txxApeM.Text.Trim());
            per.Nombres =
Metodos.LetraTipoOracion(tbxNombres.Text.Trim());
            per.FecNac = dtpFecNac.Value;
            per.Sexo =
cbxSexo.SelectedItem.ToString();
            if (ADNPersona.ExisteNomApe(per))
            {
                DialogResult res =
MessageBox.Show("Se han encontrado coincidencias de nombres
y/o apellidos.\n¿Desea Continuar?", "Aviso",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (res == DialogResult.Yes)
                {
                    GuardarDatos();
                }
            }
            else
            {
                GuardarDatos();
            }
        }
    }
}

```

```

        else
        {
            MessageBox.Show("El número de telefono o
celular no tiene el formato correcto.", "Datos", MessageBoxButtons.OK,
MessageBoxIcon.Stop);
        }
    }
    else
    {
        MessageBox.Show("Faltan Datos", "Datos",
MessageBoxButtons.OK, MessageBoxIcon.Stop);
    }
}

private void chbxFeligres_CheckedChanged(object
sender, EventArgs e)
{
    if (chbxFeligres.Checked)
    {
        btnBuscarFeligres.Visible = true;
Metodos.Habilitar(groupBox3, false);
        btnBuscarFeligres.Enabled = true;
    }
    else
    {
        btnBuscarFeligres.Visible = false;
Metodos.Habilitar(groupBox3, true);
    }
    Metodos.Limpiar(groupBox3); per.IdPersona = 0;
}

private void btnBuscarFeligres_Click(object sender,
EventArgs e)
{
    per.IdPersona = 0; invoca = true;
    frmBuscarFeligres f = new frmBuscarFeligres();
f.ShowDialog();
    if (per.IdPersona > 0)
    {
        tbxApeP.Text = per.ApePaterno; tbxApeM.Text
= per.ApeMaterno;
        tbxNombres.Text = per.Nombres; cbxSexo.Text
= per.Sexo;
        dtpFecNac.Value = per.FecNac;
    }
    else
    { per.IdPersona = 0; Metodos.Limpiar(groupBox3);
}
    invoca = false;
}

private void btnModificar_Click(object sender,
EventArgs e)
{
    var f = dgvLista.CurrentRow;
    idP = Convert.ToInt32(f.Cells[0].Value);
    per.IdPersona =
Convert.ToInt32(f.Cells[1].Value);
    tbxApeP.Text = f.Cells[2].Value.ToString();
    tbxApeM.Text = f.Cells[3].Value.ToString();
    tbxNombres.Text = f.Cells[4].Value.ToString();
}

```

```

        dtpFecNac.Value =
Convert.ToDateTime(f.Cells[5].Value);
        cbxSexo.Text = f.Cells[6].Value.ToString();
        cbxTipo.Text = f.Cells[8].Value.ToString();
        tbxDomicilio.Text = f.Cells[9].Value.ToString();
        mtbxTelefono.Text =
f.Cells[10].Value.ToString();
        mtbxCelular.Text = f.Cells[11].Value.ToString();
        btnNuevo.PerformClick();
        chbxFeligrés.Visible = false;
    }

    private void dgvLista_Click(object sender, EventArgs
e)
{
    if (dgvLista.Rows.Count > 0)
    {
        btnModificar.Enabled = true;
        btnEliminar.Enabled = true;
    }
    else
    {
        btnModificar.Enabled = false;
        btnEliminar.Enabled = false;
    }
}

private void btnEliminar_Click(object sender,
EventArgs e)
{
    string apep =
(string)dgvLista.CurrentRow.Cells[2].Value,
apem =
(string)dgvLista.CurrentRow.Cells[3].Value,
nom =
(string)dgvLista.CurrentRow.Cells[4].Value;
    DialogResult res = MessageBox.Show("¿Realmente
desea eliminar a " + apep + " " + apem + ", " +
nom + "?", "Eliminar Registro",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (res == DialogResult.Yes)
    {
        Personal p = new Personal();
        p.IdPersonal =
Convert.ToInt32(dgvLista.CurrentRow.Cells[0].Value);
        p.NomPersona = new
Persona(Convert.ToInt32(dgvLista.CurrentRow.Cells[1].Value))
;
        if (ADNPpersonal.Eliminar(p))
        {
            MessageBox.Show("Registro eliminado con
exito.", "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            VerDatos(); btnEliminar.Enabled = false;
        btnModificar.Enabled = false; per.IdPersona = 0;
        }
        else
        {
            MessageBox.Show("No se puede eliminar.",
"Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
}

```

```

        }
    }

    private void ManejadorLetras(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
}

private void ManejadorFiltro(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
    if (e.KeyChar == (char)Keys.Enter)
    {
        btnBuscar.PerformClick();
    }
}
}
}

```

- frmPrincipal.cs

```

using Utilidades;
using System.Drawing.Drawing2D;
using Entidades;
using AccesoDatosNegocio;

namespace Presentacion
{
    public partial class frmPrincipal : RibbonForm
    {
        public frmPrincipal()
        {
            InitializeComponent();
        }

        public static bool b = false, c = false, m = false;
        public static List<Departamento> lis1 = null, lis2 =
null, lis3 = null;

        private void frmPrincipal_Load(object sender,
EventArgs e)
        {
            if (ADNAudi.ContarTablas() == false)
            {
                string rutainicio =
Environment.CurrentDirectory + @"\Inicio.bak";
                Metodos.Restore(rutainicio);
            }

            lis1 = ADNDepartamento.Lista(new Pais(27));
            lis2 = new List<Departamento>(); lis3 = new
List<Departamento>();

            foreach (Departamento item in lis1)
            {
                lis2.Add(item); lis3.Add(item);
            }
        }
    }
}

```

```

                ADNAudi.Insertar(new Audi("Ingreso al
Sistema")));
            }

        private void frmPrincipal_FormClosing(object sender,
FormClosingEventArgs e)
{
    var res = MessageBox.Show("Desea Salir de
Sistema", "Cerrar", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
    if (res == DialogResult.Yes)
    {
        ADNAudi.Insertar(new Audi("Salida del
Sistema"));
        Environment.Exit(0);
    }
    else
    {
        e.Cancel = true;
    }
}

private void rbtnPaises_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmPaises>(this);
}

private void rbtnCiudades_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmDepartamentos>(this);
}

private void rbtnProvincias_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmProvincias>(this);
}

private void rbtnTemplos_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmTemplo>(this);
}

private void rbtnTipos_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmTipos>(this);
}

private void rbtnLugares_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmLugarReunion>(this);
}

private void rbtnFeligres_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmFeligres>(this);
}

```

```

        }

        private void rbtnBautismo_Click(object sender,
EventArgs e)
{
    if (ADNLibroSacramento.HayLibroAbierto(new
LibroSacramento("Bautismo")))
    {
        Metodos.ControlarForm<frmBautismo>(this);
    }
    else
    {
        MessageBox.Show("No hay ningun libro de
bautismo abierto.\nDebe abrir uno para registrar nuevos
datos",
                    "No hay libros abiertos",
                    MessageBoxButtons.OK, MessageBoxIcon.Stop);
    }
}

private void rbtnConfirmacion_Click(object sender,
EventArgs e)
{
    if (ADNLibroSacramento.HayLibroAbierto(new
LibroSacramento("Confirmacion")))
    {

        Metodos.ControlarForm<frmConfirmacion>(this);
    }
    else
    {
        MessageBox.Show("No hay ningun libro de
confirmación abierto.\nDebe abrir uno para registrar nuevos
datos",
                    "No hay libros abiertos",
                    MessageBoxButtons.OK, MessageBoxIcon.Stop);
    }
}

private void rbtnMatrimonio_Click(object sender,
EventArgs e)
{
    if (ADNLibroSacramento.HayLibroAbierto(new
LibroSacramento("Matrimonio")))
    {
        Metodos.ControlarForm<frmMatrimonio>(this);
    }
    else
    {
        MessageBox.Show("No hay ningun libro de
matrimonio abierto.\nDebe abrir uno para registrar nuevos
datos",
                    "No hay libros abiertos",
                    MessageBoxButtons.OK, MessageBoxIcon.Stop);
    }
}

private void rbtnActas_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmConsultaFeligres>(this);
}

```

```

        }

        private void rbtnLibro_Click(object sender,
EventArgs e)
{
    b = (Metodos.FormAbierto<frmBautismo>()) ? true
: false;
    c = (Metodos.FormAbierto<frmConfirmacion>()) ?
true : false;
    m = (Metodos.FormAbierto<frmMatrimonio>()) ?
true : false;
    if (!(b && c && m))
    {

Metodos.ControlarForm<frmLibroSacramento>(this);
    }
    else
    {
        MessageBox.Show("No se puede abrir la
ventana.\n" +
                    "Al menos una ventana de sacramentos
debe estar cerrada", "Accion no Permitada",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Stop);
    }
}

private void rbtnPersonal_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmPersonal>(this);
}

private void rbtnComunidad_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmComunidades>(this);
}

private void rmiSalir_Click(object sender,
EventArgs e)
{
    Application.Exit();
}

private void rbtnAsignacion_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmAsignacion>(this);
}

private void rbtnPersonalAsignado_Click(object
sender, EventArgs e)
{
    Metodos.ControlarForm<frmPersonalAsignado>(this);
}

private void rbtnBackup_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmBakcup>(this);
}

```

```

        }

        private void rbtnRestore_Click(object sender,
EventArgs e)
{
    Metodos.ControlarForm<frmRestore>(this);
}

private void frmPrincipal_Resize(object sender,
EventArgs e)
{
    this.Refresh();
}

}

```

- frmTemplo.cs

```

using Entidades;
using AccesoDatosNegocio;
using Utilidades;

namespace Presentacion
{
    public partial class frmTemplo :
Presentacion.Plantillas.frmPlantillaForm
    {
        public frmTemplo()
        {
            InitializeComponent();
        }

        static bool ColocarPro = false; static bool dis =
true;
//static List<Departamento> listadepa =
ADNDepartamento.Lista(new Pais(27));
static List<Pais> listapais = ADNPais.Lista();
static int idP = 0;

void CargarDepartamento()
{
    //cbxDepa.DataSource = listadepa;
    cbxDepa.DataSource = frmPrincipal.lis1;
    cbxDepa.DisplayMember = "Nombre";
    cbxDepa.ValueMember = "IdDepartamento";
}

void CargarPais()
{
    cbxDepa.DisplayMember = "Nombre";
    cbxDepa.DataSource = listapais;
    cbxDepa.ValueMember = "IdPais";
}

void CargarProvincia()
{
    if (ColocarPro)
    {
        if (chbxNacional.Checked)
        {

```

```

        Departamento d = new Departamento();
        d.IdDepartamento =
Convert.ToInt32(cbxDepa.SelectedValue);
        cbxPro.DataSource =
ADNProvincia.Lista(d);
        cbxPro.DisplayMember = "Nombre";
        cbxPro.ValueMember = "IdProvincia";
    }
else
{
    Pais p = new Pais();
    p.IdPais =
Convert.ToInt32(cbxDepa.SelectedValue);
    cbxPro.DataSource =
ADNDepartamento.Lista(p);
    cbxPro.DisplayMember = "Nombre";
    cbxPro.ValueMember = "IdDepartamento";
}
}

void VerDatos() {
    dgvLista.Rows.Clear();
    Templo t = new Templo(tbxFiltro.Text.Trim());
    List<Templo> lista = ADNTemplo.Lista(t,0);
    foreach (var fila in lista)
    {
        dgvLista.Rows.Add(fila.DatosToArray());
    }
}

void HabilitarDistrito(bool est) {
    if (dis)
    {
        if (est)
        {
            this.ClientSize = new
System.Drawing.Size(437, 538);
            this.groupBox3.Size = new
System.Drawing.Size(418, 181);
            btnGuardar.Location = new
System.Drawing.Point(154, 130);
            label4.Visible = true;
            cbxDistrito.Visible = true;
        }
        else
        {
            this.ClientSize = new
System.Drawing.Size(437, 501);
            this.groupBox3.Size = new
System.Drawing.Size(418, 144);
            btnGuardar.Location = new
System.Drawing.Point(269, 93);
            label4.Visible = false;
            cbxDistrito.Visible = false;
        }
    }
}

private void frmParroquia_Load(object sender,
EventArgs e)
{

```

```

        dis = false; ColocarPro = false;
CargarDepartamento(); cbxDepa.SelectedIndex = 1;
                    ColocarPro = true; cbxDepa.SelectedIndex = 0;
                    cbxDistrito.SelectedIndex = 0;
                    this.ClientSize = new System.Drawing.Size(437,
342);
                    cbxDepa.SelectedValue = 460;
                    cbxPro.SelectedValue = 102; cbxTipo.SelectedIndex = 0;
                    VerDatos();
                }

        private void btnBuscar_Click(object sender,
EventArgs e)
{
    VerDatos();
}

        private void btnCerrar_Click(object sender,
EventArgs e)
{
    this.Close();
}

        private void btnNuevo_Click(object sender, EventArgs
e)
{
    if (btnNuevo.Text == "Nuevo")
    {
        btnNuevo.Text = "Cancelar"; btnNuevo.Image =
global::Presentacion.Resource1.cancel;
        btnModificar.Enabled = false;
        btnEliminar.Enabled = false;
        this.ClientSize = new
System.Drawing.Size(437, 538); dgvLista.Enabled = false;
        Metodos.Limpiar(groupBox1);
        Metodos.Habilitar(groupBox1, false); dis = true;
    }
    else
    {
        this.ClientSize = new
System.Drawing.Size(437, 342); dis = false;
        btnNuevo.Text = "Nuevo"; btnNuevo.Image =
global::Presentacion.Resource1.Document;
        Metodos.Limpiar(groupBox3);
        Metodos.Habilitar(groupBox1, true);
        idP = 0; dgvLista.Enabled = true;
        cbxTipo.SelectedIndex = 0;
        cbxDepa.SelectedValue = 460;
        cbxPro.SelectedValue = 102;
    }
}

        private void btnModificar_Click(object sender,
EventArgs e)
{
    DataGridViewRow f = dgvLista.CurrentRow;
    idP = Convert.ToInt32(f.Cells[0].Value);
    tbxNombre.Text = f.Cells[1].Value.ToString();
    cbxTipo.Text = f.Cells[2].Value.ToString();
    if (Convert.ToInt32(f.Cells[3].Value) == 27)
    {
}
}

```

```

        chbxNacional.Checked = true;
    }
    else
    {
        chbxNacional.Checked = false;
    }
    cbxDepa.Text = f.Cells[4].Value.ToString();
    cbxPro.Text = f.Cells[5].Value.ToString();
    cbxDistrito.Text = f.Cells[6].Value.ToString();
    btnNuevo.PerformClick();
}

private void btnEliminar_Click(object sender,
EventArgs e)
{
    DialogResult res = MessageBox.Show("¿Desea
eliminar el registro?", "Eliminar Registro",
MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
    if (res==DialogResult.Yes)
    {
        int id =
Convert.ToInt32(dgvLista.CurrentRow.Cells[0].Value);
        if (ADNTemplo.Eliminar(new Templo(id)))
        {
            MessageBox.Show("Registro eliminado con
exito.", "Exito", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            VerDatos(); btnEliminar.Enabled = false;
            btnModificar.Enabled = false;
        }
        else
        {
            MessageBox.Show("No se puede eliminar.",
"Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
    }
}

private void btnGuardar_Click(object sender,
EventArgs e)
{
    if (tbxNombre.TextLength >0)
    {
        Templo t = new Templo();
        t.IdTemplo = idP; t.Nombre =
tbxNombre.Text.Trim(); t.Tipo = cbxTipo.Text;
        int iddepa = (chbxNacional.Checked) ?
Convert.ToInt32(cbxDepa.SelectedValue) :
Convert.ToInt32(cbxPro.SelectedValue);
        int idpro = (chbxNacional.Checked) ?
Convert.ToInt32(cbxPro.SelectedValue) : 0;
        t.LugarDepa = new Departamento(iddepa);
        t.LugarPro = new Provincia(idpro);
        t.Distrito = (cbxDistrito.Visible &&
cbxDistrito.Text.Length>0) ? cbxDistrito.Text : null;
        if (ADNTemplo.ExisteNombre(t))
        {
            DialogResult res = MessageBox.Show("No
pueden haber dos templos con el mismo nombre dentro de un
departamento o ciudad",

```

```

                "Duplicado", MessageBoxButtons.OK,
                MessageBoxIcon.Stop);
            }
            else
            {
                DialogResult res = (idP == 0) ?
                    MessageBox.Show("¿Los datos ingresados son correctos?", "Verificacion del Registro",
                        MessageBoxButtons.YesNo, MessageBoxIcon.Question)
                        : DialogResult.Yes;
                if (res==DialogResult.Yes)
                {
                    if (ADNTemplo.Guardar(t))
                    {
                        string r = (idP == 0) ? "guardados" : "actualizados";
                        MessageBox.Show("Datos" + r +
                            "con exito.", "Exito", MessageBoxButtons.OK,
                            MessageBoxIcon.Information);
                        btnNuevo.PerformClick();
                    }
                    VerDatos();
                }
                else
                {
                    MessageBox.Show("Ha ocurrido un problema.", "Error", MessageBoxButtons.OK,
                        MessageBoxIcon.Error);
                }
            }
        }
        else
        {
            MessageBox.Show("Faltan llenar datos.", "Faltan Datos", MessageBoxButtons.OK, MessageBoxIcon.Stop);
        }
    }

    private void dgvLista_Click(object sender, EventArgs e)
    {
        if (dgvLista.Rows.Count > 0)
        {
            if (Convert.ToInt32(dgvLista.CurrentRow.Cells[0].Value) > 1)
            {
                btnEliminar.Enabled = true;
                btnModificar.Enabled = true;
            }
            else
            {
                btnEliminar.Enabled = false;
                btnModificar.Enabled = false;
            }
        }
        else
        {
            btnModificar.Enabled = false;
            btnEliminar.Enabled = false;
        }
    }
}

```

```

        private void cbxDepa_SelectedIndexChanged(object
sender, EventArgs e)
{
    CargarProvincia();

    if (dis==true && chbxNacional.Checked &&
        cbxPro.Items.Count>0 &&
Convert.ToInt32(cbxDepa.SelectedValue) != 460)
    {
        HabilitarDistrito(false);
    }
}

private void chbxNacional_CheckedChanged(object
sender, EventArgs e)
{
    ColocarPro = false; HabilitarDistrito(false);
dis = false;
    if (chbxNacional.Checked)
    {
        lblDepa.Text = "Departamento:";
CargarDepartamento(); lblPro.Text = "Provincia:";
    }
    else
    {
        lblDepa.Text = "Pais:"; CargarPais();
lblPro.Text = "Ciudad:";
    }
    cbxDepa.SelectedIndex = 1; ColocarPro = true;
cbxDepa.SelectedIndex = 0; dis = true;
}

private void cbxTipo_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (cbxTipo.SelectedIndex==0)
    {
        if (chbxNacional.Checked &&
Convert.ToInt32(cbxDepa.SelectedValue) == 460 &&
Convert.ToInt32(cbxPro.SelectedValue) == 102)
        {
            HabilitarDistrito(true);
        }
        //cbxDepa.SelectedValue = 460;
cbxPro.SelectedValue = 102;
    }
    else
    {
        if (chbxNacional.Checked &&
cbxDistrito.Visible)
        {
            HabilitarDistrito(false);
        }
        //cbxDepa.SelectedIndex = 0;
    }
}

private void ManejadorLetras(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
}

```

```
private void tbxFiltro_KeyPress(object sender,
KeyPressEventArgs e)
{
    Metodos.SoloLetras(e);
    if (e.KeyChar == (char)Keys.Enter)
    {
        btnBuscar.PerformClick();
    }
}

private void cbxPro_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (Convert.ToInt32(cbxDepa.SelectedValue) ==
460 && Convert.ToInt32(cbxPro.SelectedValue) == 102 &&
cbxTipo.SelectedIndex==0)
    {
        HabilitarDistrito(true);
    }
    else
    {
        HabilitarDistrito(false);
    }
}
}
```

## Casos de Prueba

<b>ID CASO DE PRUEBA</b>	CP001	<b>NOMBRE</b>	Caso de Prueba de Gestión de Feligrés
<b>MÓDULO A PROBAR</b>	Módulo de Feligrés		
<b>CASO DE USO</b>	Gestión de Feligrés		
<b>REGISTRAR CORRECTAMENTE LOS DATOS DEL FELIGRÉS</b>			
<b>DESCRIPCIÓN</b>	Registrar de manera correcta los datos del feligrés en el sistema		
<b>REQUERIMIENTO(s)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos del feligrés.</li> <li>Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Feligrés” en el sistema.</li> <li>2. Hacer click en “Nuevo”.</li> <li>3. Digitar los datos personales del Feligrés.</li> <li>4. Seleccionar el lugar de nacimiento del Feligrés.</li> <li>5. Digitar los datos del padre.</li> <li>6. Seleccionar el lugar de nacimiento del padre.</li> <li>7. Digitar los datos de la madre.</li> <li>8. Seleccionar el lugar de nacimiento de la madre.</li> <li>9. Hacer click en “Guardar”.</li> <li>10. Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos del feligrés son registrados correctamente en el sistema		
<b>ESTADO</b>			
<b>MODIFICAR CORRECTAMENTE LOS DATOS DEL FELIGRÉS</b>			
<b>DESCRIPCIÓN</b>	Actualizar de manera correcta los datos del feligrés en el sistema		
<b>REQUERIMIENTO(s)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos del feligrés.</li> <li>Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El feligrés debe estar registrado en el sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Feligrés” en el sistema.</li> <li>2. Buscar al feligrés por su apellido paterno y/o materno.</li> <li>3. Seleccionar al feligrés.</li> <li>4. Hacer click en “Modificar”</li> <li>5. Modificar los datos del feligrés.</li> <li>6. Hacer click en “Guardar”.</li> <li>7. Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos del feligrés son modificados correctamente del sistema		
<b>ESTADO</b>			
<b>ELIMINAR CORRECTAMENTE EL REGISTRO DEL FELIGRÉS</b>			
<b>DESCRIPCIÓN</b>	Eliminar el registro de un feligrés del sistema		
<b>REQUERIMIENTO(s)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos del feligrés.</li> <li>Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El feligrés debe estar registrado en el sistema</li> </ul>		

<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Feligrés” en el sistema.</li> <li>2. Buscar al feligrés por su apellido paterno y/o materno.</li> <li>3. Seleccionar al feligrés.</li> <li>4. Hacer click en “Eliminar”.</li> <li>5. Confirmar la eliminación.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del feligrés son eliminados correctamente del sistema
<b>ESTADO</b>	
<b>BUSCAR RÁPIDAMENTE LOS DATOS DE UN FELIGRÉS</b>	
<b>DESCRIPCIÓN</b>	Buscar al feligrés de manera rápida
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Buscar los registros de forma rápida y precisa.</li> <li>• Ordenar los registros del feligrés y personal alfabéticamente por apellidos.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El feligrés debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Feligrés” en el sistema.</li> <li>2. Ingresar las primeras letras del apellido paterno y/o materno del feligrés, o su apellido completo.</li> <li>3. Presionar “Enter” en cualquiera de las cajas de texto de filtrado o hacer click en “Buscar”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del feligrés son visualizados rápidamente
<b>ESTADO</b>	
<b>ARRASTRAR LOS DATOS DE UN PERSONAL PARA REGISTRARLO COMO FELIGRÉS</b>	
<b>DESCRIPCIÓN</b>	Buscar al personal y arrastrar sus datos básicos.
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Posibilidad de registrar a un personal como feligrés y viceversa.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El personal debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Hacer click en el checkbox “Esta como Personal”.</li> <li>2. Hacer click en el botón “Buscar Personal”.</li> <li>3. En la nueva ventana, ingresar las primeras letras del apellido paterno y/o materno del personal, o su apellido completo.</li> <li>4. Presionar “Enter” en cualquiera de las cajas de texto de filtrado o hacer click en “Buscar”.</li> <li>5. Seleccionar el personal.</li> <li>6. Presionar “Enter”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del personal son arrastrados a la ventana para su registro como feligrés
<b>ESTADO</b>	

Tabla 26 Caso de Prueba de Gestión de Feligrés

<b>ID CASO DE PRUEBA</b>	CP002	<b>NOMBRE</b>	Caso de Prueba de Gestión de Sacramentos
<b>MODULO A PROBAR</b>	Módulo de Sacramentos		
<b>CASO DE USO</b>	Gestión de Sacramentos		
<b>REGISTRAR CORRECTAMENTE LOS DATOS DEL SACRAMENTO</b>			
<b>DESCRIPCIÓN</b>	Registrar de manera correcta los datos del sacramentos del feligrés en el sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos del Bautismo: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Natural del Padre, Nombres de la Madre, Natural de la Madre, Lugar de nacimiento, Fecha de nacimiento, Fecha de Bautismo, Nombre del Padrino, Nombre de la Madrina, Ministro o Celebrante, Nota Marginal. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.</li> <li>• Registrar los datos específicos de la Confirmación: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Nombres de la Madre, Lugar de nacimiento, Fecha de nacimiento, Lugar de Bautismo, Fecha de Bautismo, Nº Libro – Fojas – Número del Bautismo, Fecha de la Confirmación, Nombre del Padrino o Madrina, Ministro o Celebrante, Anotaciones. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.</li> <li>• Registrar los datos específicos del Matrimonio: Nombre de la Parroquia, Nº Libro – Fojas – Número, Nombres del Esposo, Nacido en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Nombres de la Esposa, Nacida en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Fecha de Bautismo, Fecha del Matrimonio, Nombre del Padrino, Nombre de la Madrina, Nombre de los Testigos (2), Ministro o Celebrante, Anotaciones. Contemplar las acciones: Nuevo, Guardar, Modificar, Buscar y Cancelar.</li> <li>• Impedir el registro de sacramentos si no hay ningún libro de dicho sacramento aperturado.</li> <li>• Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros soportados.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El feligrés debe estar registrado en el sistema</li> <li>• El templo debe estar registrado en el sistema</li> <li>• Debe haber al menos un libro aperturado del sacramento</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Bautismo” en el sistema.</li> <li>2. Hacer click en “Nuevo”.</li> <li>3. “Buscar al Feligrés” (“Esposo” y “Esposa” en el caso del Matrimonio).</li> <li>4. Ingresar los datos del sacramento.</li> <li>5. “Buscar el templo”.</li> <li>6. Seleccionar el libro, folio y número.</li> <li>7. Hacer click en “Guardar”.</li> </ol>		

	8. Confirmar los datos.
<b>RESULTADO ESPERADO</b>	Los datos del sacramento son registrados correctamente
<b>ESTADO</b>	
<b>BUSCAR AL FELIGRÉS Y ARRASTRAR SUS DATOS AL SACRAMENTO</b>	
<b>DESCRIPCIÓN</b>	Buscar el registro del feligrés para arrastrar sus datos al sacramento
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Buscar los registros de forma rápida y precisa.</li> <li>• Ordenar los registros del feligrés y personal alfabéticamente por apellidos.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El feligrés debe estar registrado en el sistema</li> <li>• Estar en proceso de registrar un sacramento</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. En la ventana de registro del sacramento, hacer click en “Buscar Feligrés” (Buscar Esposo o Esposa en el caso del matrimonio).</li> <li>2. Ingresar las primeras letras del apellido paterno y/o materno del feligrés, o su apellido completo.</li> <li>3. Presionar “Enter” en cualquiera de las cajas de texto de filtrado o hacer click en “Buscar”.</li> <li>4. Seleccionar al feligrés.</li> <li>5. Presionar “Enter”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del feligrés son arrastrados a la ventana para el registro del sacramento.
<b>ESTADO</b>	
<b>BUSCAR AL TEMPLO Y ARRASTRAR SUS DATOS AL SACRAMENTO</b>	
<b>DESCRIPCIÓN</b>	Buscar el registro del templo para arrastrar sus datos al sacramento
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Buscar los registros de forma rápida y precisa.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del Sistema</li> <li>• El templo debe estar registrado en el sistema</li> <li>• Estar en proceso de registrar un sacramento</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Hacer click en “Buscar Templo”.</li> <li>2. En la nueva ventana, ingresar las primeras letras del nombre del templo o su nombre completo.</li> <li>3. Hacer click en “Buscar” o presionar “Enter” en la caja de texto de filtrado.</li> <li>4. Seleccionar el templo.</li> <li>5. Presionar “Enter”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del templo son arrastrados a la ventana para el registro del sacramento.
<b>ESTADO</b>	
<b>MODIFICAR CORRECTAMENTE LOS DATOS DE LOS SACRAMENTOS</b>	
<b>DESCRIPCIÓN</b>	Actualizar de manera correcta los datos del bautismo
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• No permitir la modificación de los registros de sacramentos para los campos: feligrés y libro, folio, número del libro.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del Sistema</li> <li>• El sacramento debe estar registrado</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana del sacramento en el sistema.</li> <li>2. Buscar el sacramento por el apellido paterno y/o materno del feligrés.</li> <li>3. Seleccionar el registro.</li> <li>4. Hacer click en “Modificar”.</li> <li>5. Modificar los datos del sacramento.</li> </ol>

	<p>6. Hacer click en “Guardar”.</p> <p>7. Confirmar los datos.</p>
<b>RESULTADO ESPERADO</b>	Los datos del sacramento son modificados correctamente
<b>ESTADO</b>	
<b>INGRESAR LOS DATOS DEL BAUTISMO EN LA CONFIRMACIÓN Y MATRIMONIO</b>	
<b>DESCRIPCIÓN</b>	
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Posibilidad de ingresar los datos del bautismo del feligrés en la confirmación (esposo o esposa en el matrimonio) manualmente, en caso que el feligrés no tenga registros de bautismo en la parroquia.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El feligrés debe estar registrado en el sistema</li> <li>• Estar en proceso de registrar una confirmación o matrimonio</li> <li>• El bautismo del feligrés se ha efectuado en otra parroquia</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. “Buscar el Templo”</li> <li>2. Ingresar la fecha de Bautismo</li> <li>3. Ingresar el libro y folio del bautismo, en caso de la confirmación</li> </ol>
<b>RESULTADO ESPERADO</b>	Al no tener datos del bautismo del feligrés, estos datos son ingresados manualmente por el usuario.
<b>ESTADO</b>	
<b>BUSCAR RÁPIDAMENTE LOS DATOS DE UN SACRAMENTO</b>	
<b>DESCRIPCIÓN</b>	Buscar rápidamente los datos pertenecientes al registro de un sacramento del feligrés
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Buscar los registros de forma rápida y precisa.</li> <li>• Ordenar los registros del feligrés y personal alfabéticamente por apellidos.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El sacramento debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana del sacramento en el sistema.</li> <li>2. Ingresar las primeras letras del apellido paterno y/o materno del feligrés, o su apellido completo.</li> <li>3. Presionar “Enter” en cualquiera de las cajas de texto de filtrado o hacer click en “Buscar”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del sacramento son fácilmente encontrados y de manera rápida
<b>ESTADO</b>	

Tabla 27 Caso de Prueba de Gestión de Sacramentos

<b>ID CASO DE PRUEBA</b>	CP003	<b>NOMBRE</b>	Caso de Prueba de Gestión del Libro Sacramental
<b>MODULO A PROBAR</b>	Módulo de Sacramentos		
<b>CASO DE USO</b>	Gestión del Libro Sacramental		
<b>REGISTRAR(APERTURAR) UN LIBRO SACRAMENTAL</b>			
<b>DESCRIPCIÓN</b>	Registrar(Aperturar) de manera correcta los datos de un libro sacramental en el sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios, Nº de registros por folio, fecha de apertura, fecha de cierre, estado, cantidad de registros.</li> </ul> <p>Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reaperturar Libro, Buscar y Cancelar</p>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Libro Sacramental” en el sistema.</li> <li>2. Hacer click en “Nuevo”.</li> <li>3. Ingresar los datos de libro.</li> <li>4. Hacer click en “Guardar”.</li> <li>5. Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos son registrados en el sistema. El libro queda aperturado y listo para registrar sacramentos.		
<b>ESTADO</b>			
<b>MODIFICAR LOS DATOS DE LOS LIBROS</b>			
<b>DESCRIPCIÓN</b>	Actualizar de manera correcta los datos de un Libro Sacramental		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Modificar los datos de un libro sacramental, sólo si está aperturado</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El libro debe estar registrado(aperturado)</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Libro Sacramental” en el sistema.</li> <li>2. Buscar al Libro Sacramental por su tipo.</li> <li>3. Seleccionar al libro.</li> <li>4. Hacer click en “Modificar”.</li> <li>5. Modificar los datos del libro.</li> <li>6. Hacer click en “Guardar”.</li> <li>7. Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos del libro son actualizados correctamente		
<b>ESTADO</b>			
<b>CERRAR UN LIBRO SACRAMENTAL DE MANERA MANUAL</b>			
<b>DESCRIPCIÓN</b>	Cerrar manualmente un libro sacramental		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios, Nº de registros por folio, fecha de apertura, fecha de cierre, estado, cantidad de registros.</li> </ul> <p>Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reaperturar Libro, Buscar y Cancelar</p>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> </ul>		

	<ul style="list-style-type: none"> <li>• El libro debe estar registrado(aperturado)</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Libro Sacramental” en el sistema.</li> <li>2. Buscar al Libro Sacramental por su tipo.</li> <li>3. Seleccionar al libro.</li> <li>4. Hacer click en “Cerrar”.</li> <li>5. Confirmar el Cierre.</li> </ol>
<b>RESULTADO ESPERADO</b>	El libro es cerrado, sin poder agregarse más registros
<b>ESTADO</b>	
<b>REAPERTURAR UN LIBRO SACRAMENTAL</b>	
<b>DESCRIPCIÓN</b>	Reaperturar un libro sacramental
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios, Nº de registros por folio, fecha de apertura, fecha de cierre, estado, cantidad de registros. Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reaperturar Libro, Buscar y Cancelar</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El libro debe estar cerrado</li> <li>• La cantidad de registros del libro no debe ser igual a la cantidad máxima soportada</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Libro Sacramental” en el sistema.</li> <li>2. Buscar al Libro Sacramental por su tipo.</li> <li>3. Seleccionar al libro.</li> <li>4. Hacer click en “Reaperturar”.</li> <li>5. Confirmar la Reapertura.</li> </ol>
<b>RESULTADO ESPERADO</b>	El libro es reaperturado
<b>ESTADO</b>	

Tabla 28 Caso de Prueba de Gestión de Libro Sacramental

<b>ID CASO DE PRUEBA</b>	CP004	<b>NOMBRE</b>	Caso de Prueba de Gestión de Templos
<b>MÓDULO A PROBAR</b>	Módulo de Sacramentos		
<b>CASO DE USO</b>	Gestión de Templos		
<b>REGISTRAR LOS DATOS DE LOS TEMPLOS</b>			
<b>DESCRIPCIÓN</b>	Registrar correctamente los datos del templo		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Registrar los datos de los templos: Nombre, tipo, lugar. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del Sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Templos” en el sistema.</li> <li>2. Hacer click en “Nuevo”.</li> <li>3. Digitar los datos del templo.</li> <li>4. Hacer click en “Guardar”.</li> <li>5. Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos del templo son registrados correctamente en el sistema		
<b>ESTADO</b>			

MODIFICAR LOS DATOS DE LOS TEMPLOS	
<b>DESCRIPCIÓN</b>	Actualizar de manera correcta los datos del templo en el sistema
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos de los templos: Nombre, tipo, lugar.</li> </ul> <p>Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</p>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>El templo debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Templos” en el sistema.</li> <li>Buscar al templo.</li> <li>Seleccionar al templo.</li> <li>Hacer click en “Modificar”</li> <li>Modificar los datos del templo.</li> <li>Hacer click en “Guardar”.</li> <li>Confirmar los datos.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del templo son modificados correctamente del sistema
<b>ESTADO</b>	
ELIMINAR LOS DATOS DE LOS TEMPLOS	
<b>DESCRIPCIÓN</b>	Eliminar el registro de un templo del sistema
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos de los templos: Nombre, tipo, lugar.</li> </ul> <p>Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</p>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>El templo debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Templos” en el sistema.</li> <li>Buscar al templo por su nombre.</li> <li>Seleccionar al templo.</li> <li>Hacer click en “Eliminar”.</li> <li>Confirmar la eliminación.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del templo son eliminados correctamente del sistema
<b>ESTADO</b>	
BUSCAR LOS DATOS DE LOS TEMPLOS	
<b>DESCRIPCIÓN</b>	Buscar al templo de manera rápida
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Buscar los registros de forma rápida y precisa.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>El templo debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Templos” en el sistema.</li> <li>Ingresar las primeras letras del nombre del templo.</li> <li>Presionar “Enter” en la caja de texto de filtrado o hacer click en “Buscar”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del templo son visualizados rápidamente
<b>ESTADO</b>	

Tabla 29 Caso de Prueba de Gestión de Templos

<b>ID CASO DE PRUEBA</b>	CP005	<b>NOMBRE</b>	Caso de Prueba de Gestión de Personal
<b>MÓDULO A PROBAR</b>	Módulo de Personal		
<b>CASO DE USO</b>	Gestión de Personal		
<b>REGISTRAR CORRECTAMENTE LOS DATOS DEL PERSONAL</b>			
<b>DESCRIPCIÓN</b>	Registrar de manera correcta los datos del personal en el sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Personal” en el sistema.</li> <li>Hacer click en “Nuevo”.</li> <li>Digitar los datos del personal.</li> <li>Hacer click en “Guardar”.</li> <li>Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos del personal son registrados correctamente en el sistema		
<b>ESTADO</b>			
<b>MODIFICAR CORRECTAMENTE LOS DATOS DEL PERSONAL</b>			
<b>DESCRIPCIÓN</b>	Actualizar de manera correcta los datos del personal en el sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>El personal debe estar registrado en el sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Personal” en el sistema.</li> <li>Buscar al personal por su apellido paterno y/o materno.</li> <li>Seleccionar al personal.</li> <li>Hacer click en “Modificar”</li> <li>Modificar los datos del personal.</li> <li>Hacer click en “Guardar”.</li> <li>Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos del personal son modificados correctamente del sistema		
<b>ESTADO</b>			
<b>ELIMINAR CORRECTAMENTE EL REGISTRO DEL PERSONAL</b>			
<b>DESCRIPCIÓN</b>	Eliminar el registro de un personal del sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>El personal debe estar registrado en el sistema</li> </ul>		

<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Personal” en el sistema.</li> <li>2. Buscar al personal por su apellido paterno y/o materno.</li> <li>3. Seleccionar al personal.</li> <li>4. Hacer click en “Eliminar”.</li> <li>5. Confirmar la eliminación.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del personal son eliminados correctamente del sistema
<b>ESTADO</b>	
<b>BUSCAR RÁPIDAMENTE LOS DATOS DE UN PERSONAL</b>	
<b>DESCRIPCIÓN</b>	Buscar al personal de manera rápida
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Buscar los registros de forma rápida y precisa.</li> <li>• Ordenar los registros del feligrés y personal alfabéticamente por apellidos.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El personal debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Personal” en el sistema.</li> <li>2. Ingresar las primeras letras del apellido paterno y/o materno del personal, o su apellido completo.</li> <li>3. Presionar “Enter” en cualquiera de las cajas de texto de filtrado o hacer click en “Buscar”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del personal son visualizados rápidamente
<b>ESTADO</b>	
<b>ARRASTRAR LOS DATOS DE UN FELIGRÉS PARA REGISTRARLO COMO PERSONAL</b>	
<b>DESCRIPCIÓN</b>	Buscar al feligrés y arrastrar sus datos básicos.
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Posibilidad de registrar a un personal como feligrés y viceversa.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• El personal debe estar registrado en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Hacer click en el checkbox “Esta como Feligrés”.</li> <li>2. Hacer click en el botón “Buscar Feligrés”.</li> <li>3. En la nueva ventana, ingresar las primeras letras del apellido paterno y/o materno del feligrés, o su apellido completo.</li> <li>4. Presionar “Enter” en cualquiera de las cajas de texto de filtrado o hacer click en “Buscar”.</li> <li>5. Seleccionar el feligrés.</li> <li>6. Presionar “Enter”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos del feligrés son arrastrados a la ventana para su registro como personal
<b>ESTADO</b>	

Tabla 30 Caso de Prueba de Gestión de Personal

<b>ID CASO DE PRUEBA</b>	CP006	<b>NOMBRE</b>	Caso de Prueba de Gestión de Comunidades
<b>MODULO A PROBAR</b>	Módulo de Comunidades		
<b>CASO DE USO</b>	Gestión de Comunidades		
<b>REGISTRAR CORRECTAMENTE LOS DATOS DE LA COMUNIDAD</b>			
<b>DESCRIPCIÓN</b>	Registrar de manera correcta los datos de la comunidad en el sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>El tipo de la comunidad debe estar registrado en el sistema</li> <li>El lugar de reunión debe estar registrado en el sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Comunidad” en el sistema.</li> <li>Hacer click en “Nuevo”.</li> <li>Digitar los datos de la comunidad.</li> <li>Seleccionar el tipo de comunidad.</li> <li>Seleccionar el lugar de reunión.</li> <li>Hacer click en “Guardar”.</li> <li>Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos de la comunidad son registrados correctamente en el sistema		
<b>ESTADO</b>			
<b>MODIFICAR CORRECTAMENTE LOS DATOS DE LA COMUNIDAD</b>			
<b>DESCRIPCIÓN</b>	Actualizar de manera correcta los datos de la comunidad en el sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>La comunidad debe estar registrada en el sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Comunidad” en el sistema.</li> <li>Buscar a la comunidad por su nombre.</li> <li>Seleccionar a la comunidad.</li> <li>Hacer click en “Modificar”</li> <li>Modificar los datos de la comunidad.</li> <li>Hacer click en “Guardar”.</li> <li>Confirmar los datos.</li> </ol>		
<b>RESULTADO ESPERADO</b>	Los datos de la comunidad son modificados correctamente del sistema		
<b>ESTADO</b>			
<b>ELIMINAR CORRECTAMENTE EL REGISTRO DE LA COMUNIDAD</b>			
<b>DESCRIPCIÓN</b>	Eliminar el registro de una comunidad del sistema		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Registrar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar,</li> </ul>		

	Modificar, Eliminar, Buscar y Cancelar.
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>La comunidad debe estar registrada en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Comunidad” en el sistema.</li> <li>Buscar a la comunidad por su nombre.</li> <li>Seleccionar a la comunidad.</li> <li>Hacer click en “Eliminar”.</li> <li>Confirmar la eliminación.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos de la comunidad son eliminados correctamente del sistema
<b>ESTADO</b>	
<b>BUSCAR RÁPIDAMENTE LOS DATOS DE UNA COMUNIDAD</b>	
<b>DESCRIPCIÓN</b>	Buscar a la comunidad de manera rápida
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Buscar los registros de forma rápida y precisa.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>La comunidad debe estar registrada en el sistema</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Comunidad” en el sistema.</li> <li>Ingresar las primeras letras del nombre de la comunidad o su nombre completo.</li> <li>Presionar “Enter” en la caja de texto de filtrado o hacer click en “Buscar”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos de la comunidad son visualizados rápidamente
<b>ESTADO</b>	

Tabla 31 Caso de Prueba de Gestión de Comunidades

<b>ID CASO DE PRUEBA</b>	CP007	<b>NOMBRE</b>	Caso de Prueba de Gestión de Asignación
<b>MÓDULO A PROBAR</b>	Módulo de Comunidades		
<b>CASO DE USO</b>	Gestión de Asignación		
<b>REGISTRAR(ASIGNAR) CORRECTAMENTE AL PERSONAL A UNA COMUNIDAD</b>			
<b>DESCRIPCIÓN</b>	Registrar(Asignar) a un personal a una comunidad determinada		
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>Posibilidad de añadir un personal a distintas comunidades y eliminar una asignación de un personal.</li> </ul>		
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>Inicio del sistema</li> <li>La comunidad debe estar registrada en el sistema</li> <li>El personal debe estar registrado en el sistema</li> </ul>		
<b>PASOS</b>	<ol style="list-style-type: none"> <li>Abrir la ventana “Asignación” en el sistema.</li> <li>Buscar a la comunidad.</li> <li>Seleccionar la comunidad.</li> <li>Hacer click en “Asignar”.</li> <li>En la sección “Personal Disponible”, buscar al personal.</li> <li>Seleccionar al personal.</li> <li>Hacer click en el botón “&gt;”.</li> <li>Hacer click en “Guardar”.</li> </ol>		

<b>RESULTADO ESPERADO</b>	Los datos de la asignación son registrados correctamente en el sistema
<b>ESTADO</b>	
<b>ELIMINAR CORRECTAMENTE EL REGISTRO DE LA ASIGNACIÓN DEL PERSONAL</b>	
<b>DESCRIPCIÓN</b>	Eliminar el registro de una asignación del sistema
<b>REQUERIMIENTO(S)</b>	<ul style="list-style-type: none"> <li>• Posibilidad de añadir un personal a distintas comunidades y eliminar una asignación de un personal.</li> </ul>
<b>PRE REQUISITOS</b>	<ul style="list-style-type: none"> <li>• Inicio del sistema</li> <li>• La asignación del personal debe estar registrada correctamente</li> </ul>
<b>PASOS</b>	<ol style="list-style-type: none"> <li>1. Abrir la ventana “Asignación” en el sistema.</li> <li>2. Buscar a la comunidad.</li> <li>3. Seleccionar la comunidad.</li> <li>4. Hacer click en “Asignar”.</li> <li>5. En la sección “Personal Asignado”, seleccionar al personal.</li> <li>6. Hacer click en el botón “&lt;”.</li> <li>7. Hacer click en “Guardar”.</li> </ol>
<b>RESULTADO ESPERADO</b>	Los datos de la asignación son eliminados correctamente del sistema
<b>ESTADO</b>	

Tabla 32 Caso de Prueba de Gestión de Asignación

### Integración de software y pruebas de integración

El sistema se implementará en la parroquia de la siguiente manera:

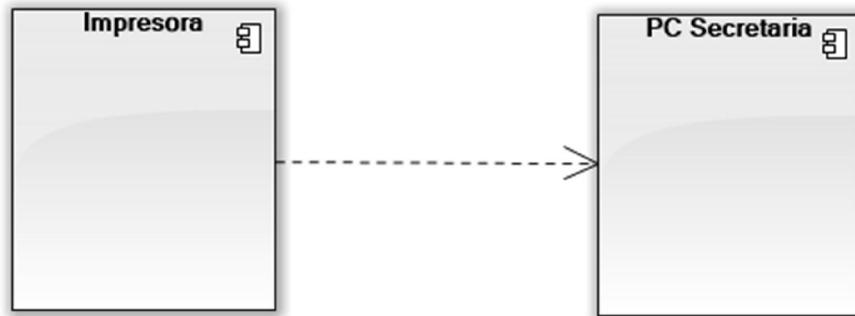


Ilustración 51 Diagrama de Despliegue

Dónde:

- PC Secretaría: La pc de escritorio donde se instalará el sistema, correspondiente a la de la secretaría de la parroquia.
- Impresora: La impresora que está conectada por cable USB a la pc de la secretaría desde donde se podrán imprimir los reportes que genera el sistema.

La pc posee sistema operativo Windows 7 Starter, el cual funciona sin ningún problema con el sistema y el motor de base de datos MySql.

Las pruebas integrales se llevarán a cabo conforme a lo siguiente:

Nº	Módulo	Acciones	Técnicas	Usuarios
1	Módulo Feligrés	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria
2	Módulo Sacramentos	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria
3	Módulo Reportes de Sacramentos	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria
4	Módulo Personal	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco
5	Módulo Comunidades	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco
6	Módulo Reportes de Asignaciones	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco
7	Módulo Backup's	Realización de copias de seguridad y posterior restauración	Pruebas Backups.	Secretaria

Tabla 33 Plan de Pruebas Integrales

## **Diccionario de Datos de Sistemas de Información**

**Sistema de Control de Sacramentos y  
Asignación de Personal**

## CONTROL DE CAMBIOS DEL DOCUMENTO

Registro de cambios

Autor	Versión	Referencia de cambios	Fecha
Juan Castillo	1.0	Original	25/10/2013

Revisores

Nombre	Versión Aprobada	Posición	Fecha
Luis Loo Parián	1.0	Asesor	04/11/2013

Propiedades del documento

Propiedad	Detalle
Título	Diccionario de Datos de Sistemas de Información
Proyecto	Sistema de Control de Sacramentos y Asignación de Personal
Autor	Juan Castillo
Nombre fichero	Diccionario de Datos
Fecha de creación	25/10/2013 10:00
Última actualización	04/11/2013 13:00

## **1. Objeto del documento**

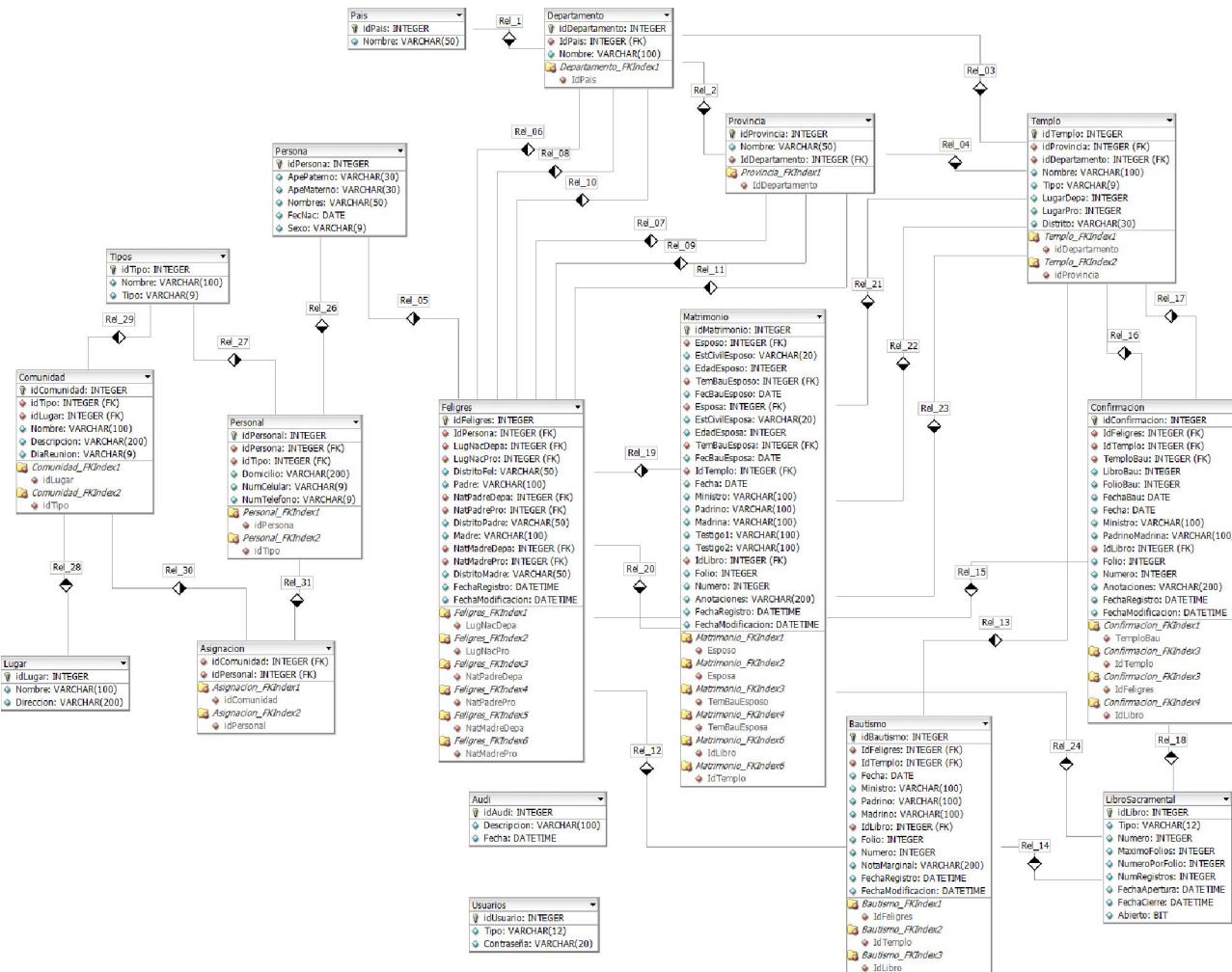
El propósito de este documento es el de ofrecer a la parroquia, el modelo de datos incluido en el sistema “Control de Sacramentos y Asignación de Personal”.

Todos los términos y tablas han sido extraídos del sistema “Control de Sacramentos y Asignación de Personal”.

## **2. Alcance del documento**

Este documento abarca todas las tablas de la base de datos del “Sistema de Control de Sacramentos y Asignación de Personal”, cada una con sus respectivas restricciones, campos y relaciones.

### 3. Diagrama Entidad-Relación



## 4. Diccionario de datos

### 4.1. Base de datos “DbParroquia”

La base de datos está compuesta por 2 partes.

La primera corresponde al control de los sacramentos que reciben los feligreses en la parroquia, en la cual se registran datos de los feligreses, los sacramentos que reciben, el libro donde se registra y el templo donde recibe el sacramento.

La segunda corresponde a la asignación del personal que hace la parroquia por medio de las distintas comunidades existentes, en la cual se registran datos del personal, las comunidades y su respectiva asignación.

#### 4.1.1. Tabla País

Base de datos DbParroquia					
Tabla País					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdPais	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967. 295	
Nombre	Nombre del País	varchar	50	A-Z , 0-9	
Restricciones					
Nombre	Descripción			Campos que afecta	
uk_pais	Clave “unique” o “única”. Los campos afectados deben ser únicos.			Nombre	

#### 4.1.2. Tabla Departamento

Base de datos DbParroquia					
Tabla Departamento					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdDepartamento	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967. 295	
Nombre	Nombre del Departamento, en el caso de Perú, o la Ciudad, en el caso de cualquier otro país.	varchar	100	A-Z , 0-9	
IdPais	Identificador del país	int		0 – 429.4967. 295	

Restricciones		
Nombre	Descripción	Campos que afecta
fk_paisdepa	Clave “foreign” o “foránea”. Enlaza a la tabla país	IdPaís
uk_departamento	Clave “unique” o “única”. Los campos afectados deben ser únicos.	Nombre, IdPaís

#### 4.1.3. Tabla Provincia

Base de datos DbParroquia					
Tabla Provincia					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdProvincia	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
Nombre	Nombre de la Provincia	varchar	50	A-Z , 0-9	
IdDepartamento	Identificador del departamento.	int		0 – 429.4967.295	
Restricciones					
Nombre	Descripción	Campos que afecta			
fk_deapro	Clave “foreign” o “foránea”. Enlaza a la tabla Departamento	IdDepartamento			
uk_provincia	Clave “unique” o “única”. Los campos afectados deben ser únicos.	Nombre, IdDepartamento			

#### 4.1.4. Tabla Templo

Base de datos DbParroquia					
Tabla Templo					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdTemplo	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
Nombre	Nombre del templo.	varchar	100	A-Z , 0-9	
Tipo	Si el templo es una Parroquia o una Capilla	varchar	9	0 – 429.4967.295	
LugarDepa	Identificador del departamento (Perú) o ciudad (Otros países).	int		0 – 429.4967.295	
LugarPro	Identificador de la Provincia.	int		0 – 429.4967.2	

				95	
Distrito	Lugar específico dentro de la provincia (OPCIONAL)	varchar	30	A-Z , 0-9	
<b>Restricciones</b>					
Nombre	<b>Descripción</b>			<b>Campos que afecta</b>	
fk_depaparr o	Clave “foreign” o “foránea”. Enlaza a la tabla Departamento			LugarDepa	
fk_proparro	Clave “foreign” o “foránea”. Enlaza a la tabla Provincia			LugarPro	
uk_parroquia	Clave “unique” o “única”. Los campos afectados deben ser únicos.			Nombre, LugarDepa	

#### 4.1.5. Tabla Persona

Base de datos DbParroquia					
Tabla Persona					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdPersona	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
ApePaterno	Apellido Paterno de la persona	varchar	30	A-Z , 0-9	
ApeMaterno	Apellido Materno de la persona	varchar	30	A-Z , 0-9	
Nombres	Nombres de la persona	varchar	50	A-Z , 0-9	
FecNac	Fecha de nacimiento de la persona	date		01/01/1001 – 31/12/9999	
Sexo	Sexo de la persona	varchar	9	A-Z , 0-9	
<b>Restricciones</b>					
Nombre	<b>Descripción</b>			<b>Campos que afecta</b>	

#### 4.1.6. Tabla Feligrés

Base de datos DbParroquia					
Tabla Feligrés					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdFeligrés	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
IdPersona	Identificador de	int		0 –	

	la persona.			429.4967. 295	
LugNacDepa	Identificador del departamento (Perú) o ciudad (Otros países).	int		0 – 429.4967. 295	
LugNacPro	Identificador de la Provincia.	int		0 – 429.4967. 295	
DistritoFel	Distrito de nacimiento del feligrés (OPCIONAL)	varchar	50	A-Z , 0-9	
Padre	Nombres y Apellidos del padre del feligrés	varchar	100	A-Z , 0-9	
NatPadreDe pa	Identificador del departamento (Perú) o ciudad (Otros países).	int		0 – 429.4967. 295	
NatPadrePr o	Identificador de la Provincia.	int		0 – 429.4967. 295	
DistritoPadr e	Distrito de nacimiento del padre (OPCIONAL)	varchar	50	A-Z , 0-9	
Madre	Nombres y Apellidos de la madre del feligrés	varchar	100	A-Z , 0-9	
NatMadreDe pa	Identificador del departamento (Perú) o ciudad (Otros países).	int		0 – 429.4967. 295	
NatMadrePr o	Identificador de la Provincia.	int		0 – 429.4967. 295	
DistritoMadr e	Distrito de nacimiento del padre (OPCIONAL)	varchar	50	A-Z , 0-9	
FechaRegistro	Fecha y hora en que se registró el feligrés	datetime		01/01/100 1 – 31/12/999 9	
FechaModificacion	Fecha y hora en que se modificaron los datos del feligrés	datetime		01/01/100 1 – 31/12/999 9	Cuando se registra el feligrés, la fecha de registro y la de modificación es la misma
<b>Restricciones</b>					
Nombre	Descripción		Campos que afecta		
fk_perfel	Clave “foreign” o “foránea”.		IdPersona		

	Enlaza a la tabla Persona	
fk_deafel	Clave “foreign” o “foránea”. Enlaza a la tabla Departamento	LugNacDepa
fk_profel	Clave “foreign” o “foránea”. Enlaza a la tabla Provincia	LugNacPro
fk_depapadre	Clave “foreign” o “foránea”. Enlaza a la tabla Departamento	NatPadreDepa
fk_propadre	Clave “foreign” o “foránea”. Enlaza a la tabla Provincia	NatPadrePro
fk_depamadre	Clave “foreign” o “foránea”. Enlaza a la tabla Departamento	NatMadreDepa
fk_promadre	Clave “foreign” o “foránea”. Enlaza a la tabla Provincia	NatMadrePro

#### 4.1.7. Tabla LibroSacramento

Base de datos DbParroquia					
Tabla LibroSacramento					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdLibro	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
Tipo	Tipo de Sacramento del Libro (Bautismo, Confirmación, Matrimonio)	int		0 – 429.4967.295	
Numero	Número del libro	int		0 – 429.4967.295	
MaximoFolios	Cantidad de folios que posee el libro	int		0 – 429.4967.295	
NumeroPorFolio	Cantidad de números por cada folio	int		0 – 429.4967.295	
NumRegistros	Cantidad de registros llenados en el libro	int		0 – 429.4967.295	
FechaApertura	Fecha en que se apertura el libro	datetime		01/01/1001 – 31/12/9999	
FechaCierre	Fecha en que se cierra el libro	datetime		01/01/1001 – 31/12/9999	
Abierto	Estado del libro (Abierto, Cerrado)	bit		0 - 1	
Restricciones					
Nombre	Descripción			Campos que afecta	

#### 4.1.8. Tabla Bautismo

Base de datos DbParroquia					
Tabla Bautismo					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdBautismo	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
IdFeligrés	Identificador del feligrés.	int		0 – 429.4967.295	
IdTemplo	Identificador del templo.	int		0 – 429.4967.295	
Fecha	Fecha de bautismo	date		01/01/100 1 – 31/12/999 9	
Ministro	Nombre del ministro que realiza el bautismo	varchar	100	A-Z , 0-9	
Padrino	Nombres y Apellidos del Padrino	varchar	100	A-Z , 0-9	
Madrina	Nombres y Apellidos del Madrina	varchar	100	A-Z , 0-9	
IdLibro	Identificador del libro sacramental	int		0 – 429.4967.295	
Folio	Folio del libro de bautismo	int		0 – 429.4967.295	
Numero	Número de registro del libro de bautismo	int		0 – 429.4967.295	
NotaMarginal	Observaciones del Bautismo	varchar	200	A-Z , 0-9	
FechaRegistro	Fecha y hora en que se registró el feligrés	datetime		01/01/100 1 – 31/12/999 9	
FechaModificación	Fecha y hora en que se modificaron los datos del feligrés	datetime		01/01/100 1 – 31/12/999 9	Cuando se registra el feligrés, la fecha de registro y la de modificación es la misma
Restricciones					
Nombre	Descripción		Campos que afecta		
fk_perbau	Clave "foreign" o "foránea". Enlaza a la tabla Feligrés		IdFeligrés		

fk_tembau	Clave “foreign” o “foránea”. Enlaza a la tabla Templo	IdTemplo
fk_librosacra bau	Clave “foreign” o “foránea”. Enlaza a la tabla LibroSacramental	IdLibro
uk_bautismo	Clave “unique” o “única”. Los campos afectados deben ser únicos.	IdLibro, Folio, Numero

#### 4.1.9. Tabla Confirmación

Base de datos DbParroquia						
Tabla Confirmación						
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones	
IdConfirmacion	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967. 295		
IdFeligrés	Identificador del feligrés	int		0 – 429.4967. 295		
IdTemplo	Identificador del templo	int		0 – 429.4967. 295		
TemploBau	Identificador del templo	int		0 – 429.4967. 295		
LibroBau	Número del libro de bautismo del feligrés	int		0 – 429.4967. 295	Este número puede o no estar registrado, ya que el feligrés pudo haberse bautizado en otra parroquia.	
FolioBau	Número de folio del libro de bautismo	int		0 – 429.4967. 295		
FechaBau	Fecha en que el feligrés se bautizo	date		01/01/100 1 – 31/12/999 9		
Fecha	Fecha de la confirmación	date		01/01/100 1 – 31/12/999 9		
Ministro	Nombre del ministro (obispo) que realiza la confirmación	varchar	100	A-Z , 0-9		
PadrinoMadrina	Nombres y Apellidos del padrino o madrina	varchar	100	A-Z , 0-9		
IdLibro	Identificador del	int		0 –		

	libro sacramental			429.4967.295	
Folio	Folio del libro de bautismo	int		0 – 429.4967.295	
Numero	Número de registro del libro de confirmación	int		0 – 429.4967.295	
Anotaciones	Observaciones de la Confirmación	varchar	200	A-Z , 0-9	
FechaRegistro	Fecha y hora en que se registró el feligrés	datetime		01/01/1001 – 31/12/9999	
FechaModificación	Fecha y hora en que se modificaron los datos del feligrés	datetime		01/01/1001 – 31/12/9999	Cuando se registra el feligrés, la fecha de registro y la de modificación es la misma
<b>Restricciones</b>					
Nombre	Descripción	Campos que afecta			
fk_felcon	Clave “foreign” o “foránea”. Enlaza a la tabla Feligres	IdFeligres			
fk_temcon1	Clave “foreign” o “foránea”. Enlaza a la tabla Templo	IdTemplo			
fk_temcon2	Clave “foreign” o “foránea”. Enlaza a la tabla Templo	TemploBau			
fk_librosacram	Clave “foreign” o “foránea”. Enlaza a la tabla LibroSacramento	IdLibro			
uk_confirmation	Clave “unique” o “única”. Los campos afectados deben ser únicos.	IdLibro, Folio, Numero			

#### 4.1.10. Tabla Matrimonio

Base de datos DbParroquia					
Tabla Matrimonio					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdMatrimonio	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
Esposo	Identificador del feligrés	int		0 – 429.4967.295	
EstCivilEsposo	Estado civil del esposo	varchar	20	A-Z , 0-9	
EdadEsposo	Edad del esposo	int		0 – 429.4967.295	Edad en que se casa

TemBauEsposo	Identificador del templo del bautismo del esposo	int		0 – 429.4967. 295	
FecBauEsposo	Fecha de bautismo del esposo	date		01/01/100 1 – 31/12/999 9	
Esposa	Identificador del feligrés	int		0 – 429.4967. 295	
EstCivilEsposa	Estado civil de la esposa	varchar	20	A-Z , 0-9	
EdadEsposa	Edad de la esposa	int		0 – 429.4967. 295	Edad en que se casa
TemBauEsposa	Identificador del templo del bautismo de la esposa	int		0 – 429.4967. 295	
FecBauEsposa	Fecha de bautismo de la esposa	date		01/01/100 1 – 31/12/999 9	
IdTemplo	Identificador del templo donde se confirma	int		0 – 429.4967. 295	
Fecha	Fecha de la confirmación	date		01/01/100 1 – 31/12/999 9	
Ministro	Nombre del ministro que realiza el matrimonio	varchar	100	A-Z , 0-9	
Padrino	Nombres y Apellidos del padrino	varchar	100	A-Z , 0-9	
Madrina	Nombres y Apellidos de la madrina	varchar	100	A-Z , 0-9	
Testigo1	Nombres y Apellidos del primer testigo	varchar	100	A-Z , 0-9	
Testigo2	Nombres y Apellidos del segundo testigo	varchar	100	A-Z , 0-9	
IdLibro	Identificador del libro sacramental	int		0 – 429.4967. 295	
Folio	Folio del libro de bautismo	int		0 – 429.4967. 295	
Numero	Número de registro del libro de matrimonio	int		0 – 429.4967. 295	
Anotaciones	Observaciones de la Confirmación	varchar	200	A-Z , 0-9	

FechaRegistro	Fecha y hora en que se registró el feligrés	datetime		01/01/100 1 – 31/12/999 9	
FechaModificación	Fecha y hora en que se modificaron los datos del feligrés	datetime		01/01/100 1 – 31/12/999 9	Cuando se registra el feligrés, la fecha de registro y la de modificación es la misma
<b>Restricciones</b>					
Nombre	<b>Descripción</b>			<b>Campos que afecta</b>	
fk_felmatri1	Clave “foreign” o “foránea”. Enlaza a la tabla Feligrés			Esposo	
fk_felmatri2	Clave “foreign” o “foránea”. Enlaza a la tabla Feligrés			Esposa	
fk_temmatr1	Clave “foreign” o “foránea”. Enlaza a la tabla Templo			TemBauEsposo	
fk_temmatr2	Clave “foreign” o “foránea”. Enlaza a la tabla Templo			TemBauEsposa	
fk_temmatr1	Clave “foreign” o “foránea”. Enlaza a la tabla Templo			IdTemplo	
fk_librosacramat	Clave “foreign” o “foránea”. Enlaza a la tabla LibroSacramental			IdLibro	
uk_matrimonio	Clave “unique” o “única”. Los campos afectados deben ser únicos.			IdLibro, Folio, Numero	

#### 4.1.11. Tabla Tipos

Base de datos DbParroquia					
Tabla Tipos					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdTipo	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967. 295	
Nombre	Nombre del tipo	varchar	100	A-Z , 0-9	
Tipo	Si es un tipo de comunidad o de personal	varchar	9	A-Z , 0-9	
<b>Restricciones</b>					
Nombre	<b>Descripción</b>			<b>Campos que afecta</b>	

#### 4.1.12. Tabla Personal

Base de datos DbParroquia						
Tabla Personal						
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones	
IdPersonal	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295		
IdPersona	Identificador de la persona	int		0 – 429.4967.295		
IdTipo	Identificador del tipo	int		0 – 429.4967.295		
Domicilio	Dirección de vivienda (OPCIONAL)	varchar	200	A-Z , 0-9		
NumCelular	Número de celular (OPCIONAL)	varchar	9	A-Z , 0-9		
NumTelefono	Número de teléfono (OPCIONAL)	varchar	9	A-Z , 0-9		
Restricciones						
Nombre	Descripción			Campos que afecta		
fk_personapersonal	Clave “foreign” o “foránea”. Enlaza a la tabla Persona			IdPersona		
fk_tipopersonal	Clave “foreign” o “foránea”. Enlaza a la tabla Tipos			IdTipo		

#### 4.1.13. Tabla Lugar

Base de datos DbParroquia						
Tabla Lugar						
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones	
IdPersona	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295		
ApePaterno	Apellido Paterno de la persona	varchar	30	A-Z , 0-9		
ApeMaterno	Apellido Materno de la persona	varchar	30	A-Z , 0-9		
Nombres	Nombres de la persona	varchar	50	A-Z , 0-9		
FecNac	Fecha de nacimiento de la persona	date		01/01/1001 – 31/12/9999		
Sexo	Sexo de la persona	varchar	9	A-Z , 0-9		

Restricciones		
Nombre	Descripción	Campos que afecta
uk_lugarreunion	Clave “unique” o “única”. Los campos afectados deben ser únicos.	Nombre

#### 4.1.14. Tabla Comunidad

Base de datos DbParroquia					
Tabla Comunidad					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdComunidad	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
IdTipo	Identificador del tipo	int		0 – 429.4967.295	
IdLugar	Identificador del lugar	int		0 – 429.4967.295	
Nombre	Nombres de la persona	varchar	100	A-Z , 0-9	
Descripcion	Fecha de nacimiento de la persona	varchar	200	A-Z , 0-9	
DiaReunion	Sexo de la persona	varchar	9	A-Z , 0-9	
Restricciones					
Nombre	Descripción	Campos que afecta			
fk_tipocomunidad	Clave “foreign” o “foránea”. Enlaza a la tabla Tipos	IdTipo			
fk_lugarcomunidad	Clave “foreign” o “foránea”. Enlaza a la tabla Lugar	IdLugar			

#### 4.1.15. Tabla Asignación

Base de datos DbParroquia					
Tabla Comunidad					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
IdComunidad	Identificador de la comunidad	int		0 – 429.4967.295	
IdPersonal	Identificador del personal	int		0 – 429.4967.295	
Restricciones					

Nombre	Descripción	Campos que afecta
fk_comunidad_asignacion	Clave “foreign” o “foránea”. Enlaza a la tabla Comunidad	IdComunidad
fk_personal_asignacion	Clave “foreign” o “foránea”. Enlaza a la tabla Personal	IdPersonal

#### 4.1.16. Tabla Audi

Base de datos DbParroquia					
Tabla Comunidad					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
idaudi	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
descripcion	Descripción de la acción que se realizó	varchar	100	A-Z , 0-9	
fecha	Fecha y hora en que se realizó la acción	datetime		01/01/1001 – 31/12/9999	
Restricciones					
Nombre	Descripción			Campos que afecta	

#### 4.1.17. Tabla Usuarios

Base de datos DbParroquia					
Tabla Comunidad					
Nombre	Descripción conceptual	Tipo	Long	Rango de Valores	Observaciones
idusuario	Clave primaria de la tabla que identifica al registro	int		0 – 429.4967.295	
tipo	Tipo de usuario	varchar	12	A-Z , 0-9	Puede ser Administrador u Operador
contraseña	Contraseña del usuario	varchar	20	A-Z , 0-9	
Restricciones					
Nombre	Descripción			Campos que afecta	

**D. Transición**  
**Pruebas de aceptación del cliente**

**Datos de Lugar y Fecha**

<b>Fecha / Hora:</b>	Viernes 19 de Julio del 2013 / 5:00 p.m.
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco
<b>Referencia</b>	ACTA DE PRUEBAS DEL MODULO DE FELIGRES

**Participantes**

Nombre	Área (Usuaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

**Objetivo del Documento**

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas del “Módulo de Feligrés”, el cual forma parte del sub sistema de Control de Sacramentos. Asimismo se indican las aprobaciones realizadas por el usuario.

**Observaciones**

A continuación se detalla cada uno de los requerimientos que se dan solución.

Item:	Requerimiento	Estado	
		APROBADO	DESAPROBADO
1	Actualizar los datos de los países. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
2	Actualizar los datos de las ciudades o departamentos. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
3	Actualizar los datos de las provincias del Perú. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	

4	Para los países, ciudades o departamento y provincias, sólo se podrán eliminar los nuevos registros que se ingresen en el sistema.	<input checked="" type="checkbox"/>	
5	Registrar los datos del feligrés. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.		<input checked="" type="checkbox"/>
6	Posibilidad de registrar a un personal como feligrés y viceversa.	<input checked="" type="checkbox"/>	
7	Ordenar los registros del feligrés y personal alfabéticamente por apellidos.	<input checked="" type="checkbox"/>	

### Conclusiones

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	

Luisa Cajo	
------------	------------------------------------------------------------------------------------

### Datos de Lugar y Fecha

Fecha / Hora:	Viernes 02 de Agosto del 2013 / 6:00 p.m.
Lugar	Secretaría de la Parroquia San Clemente - Pisco
Referencia	ACTA DE PRUEBAS DEL MODULO DE FELIGRES N°2

### Participantes

Nombre	Área (Usaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

### Objetivo del Documento

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas del “Módulo de Feligrés”, el cual forma parte del sub sistema de Control de Sacramentos. Asimismo se indican las aprobaciones realizadas por el usuario de los requerimientos que no pasaron la primera prueba de calidad.

### Observaciones

Para esta prueba se añadieron los siguientes campos:

- Distrito del Feligrés
- Distrito del Padre
- Distrito de la Madre

A continuación se detalla cada uno de los requerimientos que se dan solución.

Item:	Requerimiento	Estado	
		APROBADO	DESAPROBADO
1	Registrar los datos del feligrés. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.		

### **Conclusiones**

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### **Aceptación Final**

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

### **Datos de Lugar y Fecha**

<b>Fecha / Hora:</b>	Miércoles 28 de Agosto del 2013 / 5:00 p.m.
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco
<b>Referencia</b>	ACTA DE PRUEBAS DE LOS MODULOS DE SACRAMENTOS Y REPORTES

### **Participantes**

Nombre	Área (Usuaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto

Vicente Diez	Párroco
Luisa Cajo	Secretaria

### Objetivo del Documento

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas de los módulos de "Sacramentos" y "Reportes de Sacramentos", el cual forma parte del sub sistema de Control de Sacramentos. Asimismo se indican las aprobaciones realizadas por el usuario.

### Observaciones

A continuación se detalla cada uno de los requerimientos que se dan solución.

Item:	Requerimiento	Estado	
		APROBADO	DESAPROBADO
1	Registrar los datos específicos del Bautismo: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Natural de, Nombres de la Madre, Natural de, Lugar de nacimiento, Fecha de nacimiento, Fecha de Bautismo, Nombre del Padrino, Nombre de la Madrina, Ministro o Celebrante, Nota Marginal.	✓	
2	Registrar los datos específicos de la Confirmación: Nombre de la Parroquia, Nº Libro – Fojas – Número, Apellidos y Nombres, Nombres del Padre, Nombres de la Madre, Lugar de nacimiento, Fecha de nacimiento, Lugar de Bautismo, Fecha de Bautismo, Nº Libro – Fojas – Número del Bautismo, Fecha de la Confirmación, Nombre del Padrino o Madrina, Ministro o Celebrante, Anotaciones.	✓	
3	Registrar los datos específicos del Matrimonio: Nombre de la Parroquia, Nº Libro – Fojas – Número, Nombres del Esposo, Nacido en, Edad, Nombres del Padre, Nombres de la Madre, Lugar de Bautismo, Nombres de la Esposa, Nacida en, Edad, Nombres del Padre, Nombres de la Madre,	✓	

	Lugar de Bautismo, Fecha de Bautismo, Fecha del Matrimonio, Nombre del Padrino, Nombre de la Madrina, Nombre de los Testigos (2), Ministro o Celebrante, Anotaciones.		
4	Registrar los datos de los templos: Nombre, tipo, lugar. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	<input checked="" type="checkbox"/>	
5	Registrar los datos de los libros sacramentales: Tipo de Sacramento, Número de Libro, Nº máximo de folios, Nº de registros por folio, fecha de apertura, fecha de cierre, estado, cantidad de registros. Contemplar las acciones: Nuevo, Guardar, Modificar, Cerrar y Reaperturar Libro, Buscar y Cancelar.	<input checked="" type="checkbox"/>	
6	Permitir la reapertura de un libro sacramental si el total de registros no es igual a la cantidad máxima soportada.	<input checked="" type="checkbox"/>	
7	Modificar los datos de un libro sacramental, sólo si está aperturado.	<input checked="" type="checkbox"/>	
8	Impedir el registro de sacramentos si no hay ningún libro de dicho sacramento aperturado.	<input checked="" type="checkbox"/>	
9	No permitir la modificación de los registros de sacramentos para los campos: feligrés y libro, folio, número del libro.	<input checked="" type="checkbox"/>	
10	Buscar los sacramentos por el Apellido Paterno del feligrés.	<input checked="" type="checkbox"/>	

11	Cerrar el libro sacramental automáticamente si este ha llegado a la cantidad máxima de registros soportados.	✓	
12	Imprimir los datos de los sacramentos con el formato de actas de la parroquia	✓	

### Conclusiones

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

### Datos de Lugar y Fecha

Fecha / Hora:	Lunes 23 de Septiembre del 2013 / 6:00 p.m.
Lugar	Secretaría de la Parroquia San Clemente - Pisco
Referencia	ACTA DE PRUEBAS DEL MODULO DE PERSONAL

## Participantes

Nombre	Área (Usaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajó	Secretaria

## Objetivo del Documento

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas del "Módulo de Personal", el cual forma parte del sub sistema de Asignación de Personal. Asimismo se indican las aprobaciones realizadas por el usuario.

## Observaciones

A continuación se detalla cada uno de los requerimientos que se dan solución.

Item:	Requerimiento	Estado	
		APROBADO	DESAPROBADO
1	Registrar los datos específicos del Personal: Apellidos y Nombres, Fecha de Nacimiento, Sexo, Domicilio, Nº de Celular, Nº de Teléfono, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
2	Actualizar los datos de tipos de personal y de comunidades. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
3	Posibilidad de registrar a un personal como feligrés y viceversa.	✓	

## Conclusiones

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

### Datos de Lugar y Fecha

Fecha / Hora:	Viernes 04 de Octubre del 2013 / 5:00 p.m.
Lugar	Secretaría de la Parroquia San Clemente - Pisco
Referencia	ACTA DE PRUEBAS DEL MODULO DE COMUNIDADES

### Participantes

Nombre	Área (Usaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

### **Objetivo del Documento**

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas del "Módulo de Personal", el cual forma parte del sub sistema de Asignación de Personal. Asimismo se indican las aprobaciones realizadas por el usuario.

### **Observaciones**

A continuación se detalla cada uno de los requerimientos que se dan solución.

Item:	Requerimiento	Estado	
		APROBADO	DESAPROBADO
1	Registrar los datos específicos de las comunidades: Nombre, Descripción, Ubicación, Día de Reunión, Lugar de Reunión, Tipo. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
2	Actualizar los datos de tipos de personal y de comunidades. Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
3	Actualizar los datos de los lugares de reunión de las comunidades Contemplar las acciones: Nuevo, Guardar, Modificar, Eliminar, Buscar y Cancelar.	✓	
4	Posibilidad de añadir un personal a distintas comunidades y eliminar una asignación de un personal.	✓	

### **Conclusiones**

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### **Aceptación Final**

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

### Datos de Lugar y Fecha

Fecha / Hora:	Miércoles 16 de Octubre del 2013 / 6:30 p.m.
Lugar	Secretaría de la Parroquia San Clemente - Pisco
Referencia	ACTA DE PRUEBAS DE LOS MODULOS DE REPORTES DE ASIGNACION Y BACKUP'S

### Participantes

Nombre	Área (Usuaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

### Objetivo del Documento

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas de los módulos de “Reportes de Asignación” y “Backup’s”, el cual forma parte del sub sistema de Control de Sacramentos. Asimismo se indican las aprobaciones realizadas por el usuario.

### **Observaciones**

A continuación se detalla cada uno de los requerimientos que se dan solución.

Item:	Requerimiento	Estado	
		APROBADO	DESAPROBADO
1	Imprimir la lista de personal asignado a una comunidad.	✓	
2	Realizar copias de seguridad de la base de datos del sistema	✓	
3	Restablecer las copias de seguridad generadas del sistema	✓	

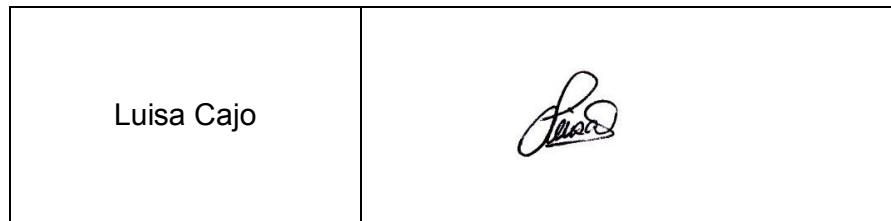
### **Conclusiones**

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### **Aceptación Final**

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	



### **Despliegue del software**

Al estar en una sola PC (la de la secretaria) el sistema no tiene un despliegue muy grande.

La PC se conecta con una impresora, la cual permitirá imprimir en formato físico los reportes que el sistema genere en PDF.

El diagrama de despliegue es el siguiente:

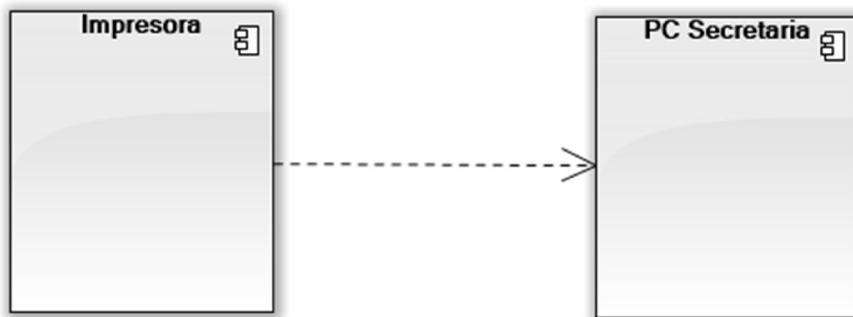


Ilustración 52 Diagrama de Despliegue

### **Plan de capacitación**

A continuación se describe, mediante un cuadro, el plan de capacitación que se seguirá para que el usuario conozca el funcionamiento del sistema.

Tema	Contenidos	Acciones	Técnicas	Dirigido a	Horas
1	Explicación detallada de las funciones del módulo Feligrés	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria	1 h.
2	Explicación	Explicación	Asociación	Secretaria	3 h.

	detallada de las funciones del módulo Sacramentos	visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	del manejo manual existente con el manejo del sistema. Pruebas de registros.		
3	Explicación detallada de las funciones del módulo Reportes de Sacramentos	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria	30 min.
4	Explicación detallada de las funciones del módulo Personal	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco	1 h.
5	Explicación detallada de las funciones del módulo Comunidades	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco	1 h.
6	Explicación detallada de las funciones del módulo Reportes de Asignaciones	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco	30 min.
7	Explicación detallada de las funciones del módulo Backup's	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Pruebas de backup's.	Secretaria	1 h.

Tabla 34 Plan de Capacitación de Usuarios

#### 4.3. Soporte del proyecto

##### 4.3.1. Gestión de la configuración

Periodo del 03-07-2013 al 31-07-2013

ESTADO DE AVANCE DE LOS ENTREGABLES				
FASE	ENTREGABLE	ESTADO DE AVANCE	ESTADO DE PRUEBAS	OBSERVACIONES
Construcción	Módulo de Feligrés	Se completó con el desarrollo del primer módulo.	Las pruebas arrojaron falencias en ciertos campos, por lo que se encuentra en correcciones.	Para obtener la aprobación del módulo, se deben agregar ciertos campos que no se tomaron en cuenta al establecer los requerimientos del software.
	Módulo de Sacramentos	A la fecha, el desarrollo del módulo se encuentra en un 45 % aproximadamente.	Las pruebas hechas en el desarrollo indican dificultades para cumplir con los requerimientos.	Aunque existen dificultades en el desarrollo, la similitud del funcionamiento de los elementos del módulo con los anteriores implicaría que se podría adelantar la fecha de entrega del módulo.

Periodo del 01-08-2013 al 30-08-2013

ESTADO DE AVANCE DE LOS ENTREGABLES				
FASE	ENTREGABLE	ESTADO DE AVANCE	ESTADO DE PRUEBAS	OBSERVACIONES
Construcción	Módulo de Sacramentos	Se completó con el desarrollo del segundo módulo.	Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	El desarrollo del módulo pudo terminarse antes de la fecha acordada, lo cual dio lugar al adelanto de fechas de desarrollo del siguiente módulo.
	Módulo de Reportes de Sacramentos	Se completó con el desarrollo del tercer módulo.	Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	Se adelantaron las fechas de desarrollo del módulo, lo cual permitió que se revisara el código fuente en busca de inconsistencias y optimización.

Periodo del 31-08-2013 al 27-09-2013

ESTADO DE AVANCE DE LOS ENTREGABLES				
FASE	ENTREGABLE	ESTADO DE AVANCE	ESTADO DE PRUEBAS	OBSERVACIONES
Construcción	Módulo de Personal	Se completó con el desarrollo del cuarto módulo.	Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	Se cumplió sin problemas con las aprobaciones del módulo.
	Módulo de Comunidades	Se sigue con el desarrollo del módulo. A la fecha se lleva un avance del 95% aproximadamente.	Las pruebas hechas en el desarrollo indican que no hay dificultades para el desarrollo del módulo.	A poco tiempo de terminar el desarrollo del módulo se estima que hay posibilidades de adelantar el desarrollo del siguiente módulo.

Periodo del 27-09-2013 al 25-10-2013

ESTADO DE AVANCE DE LOS ENTREGABLES				
FASE	ENTREGABLE	ESTADO DE AVANCE	ESTADO DE PRUEBAS	OBSERVACIONES
Construcción	Módulo de Comunidades	Se completó con el desarrollo del quinto módulo.	Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	El desarrollo del módulo pudo terminarse antes de la fecha acordada, lo cual dio lugar al adelanto de fechas de desarrollo del siguiente módulo.
	Módulo de Reportes de Asignación	Se completó con el desarrollo del sexto módulo	Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	Se adelantaron las fechas de desarrollo del módulo, lo cual permitió que se revisara el código fuente en busca de inconsistencias y optimización.

	Módulo de Backups	Se completó con el desarrollo del séptimo módulo.	Las pruebas fueron satisfactorias.	Se adelantaron las fechas de desarrollo del módulo, lo cual permitió que se revisara el código fuente en busca de inconsistencias y optimización.
--	-------------------	---------------------------------------------------	------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------

#### 4.3.2. Aseguramiento de la calidad

##### Formato de Pruebas Unitarias

<b>Módulo</b>	Feligrés
<b>Desarrollador</b>	Juan Castillo
<b>Fecha</b>	02/08/2013

#	Descripción de la prueba	Número de errores detectados	Estado
---	--------------------------	------------------------------	--------

Nota: Si no se detectan errores, colocar '0'

<i>Interfaz</i>			
1	El diseño de la pantalla / página, <u>creada o modificada</u> sigue el estándar definido en el proyecto	0	OK
2	Al hacer resize, maximize, etc., no se altera la apariencia de la pantalla.		No aplica
3	Los campos en pantalla se habilitan y deshabilitan correctamente, según la transacción o estado de la pantalla.	0	OK
4	La longitud de los campos en pantalla corresponde a la de los campos en base de datos.	0	OK
5	Los tipos de dato de los campos en pantalla corresponden a los definidos en base de datos.	0	OK
6	Los datos se ingresan, muestran y almacenan en mayúsculas o minúsculas, según lo definido.	0	OK
7	La validación de campos mandatorios funciona correctamente para todos los casos.	0	OK
8	Los campos numéricos soportan decimales, negativos, o ceros según lo definido.	0	OK
9	Los campos de fecha validan correctamente el formato definido y el rango soportado por el lenguaje de programación y por el motor de base de datos.	0	OK
10	El desplazamiento por los componentes de pantalla con la tecla TAB es ordenado.	0	OK

11	Al cancelar o grabar una transacción, los campos en pantalla se inicializan correctamente.	0	OK
12	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica
<b>Reportes</b>			
13	La longitud de los campos de salida del reporte soporta el tamaño de todos los datos definidos.	0	OK
14	El reporte se puede generar correctamente en todos los formatos definidos (Excel, texto, pdf, etc.).	0	OK
15	El formato de reporte mantiene el estándar definido por el proyecto.	0	OK
16	El reporte se puede generar correctamente con la amplitud máxima de los filtros(es decir procesando la mayor cantidad de información que permitan los filtros mandatorios.)	0	OK
17	El reporte se puede generar correctamente ingresando un valor en todos los filtros incluso los no mandatorios.		No aplica
<b>Base de Datos</b>			
18	Los objetos nuevos (SP, Tablas, Funciones, etc.) siguen los estándares de nomenclatura definidos para el proyecto.	0	OK
19	Las consultas nuevas o modificadas han sido probadas con un volumen de información similar al del ambiente de producción. En caso de ser así, ¿se comportan según lo esperado para el proyecto (en tiempo y recursos)?		No aplica
20	Se controla o tolera si el resultado de la consulta a base de datos es vacío(aun cuando la tabla debería contener información en condiciones normales)		No aplica
<b>Otros</b>			
21	El código compila sin warnings asociados al módulo u opción probada.	0	OK
22	Los flags utilizados corresponden a 0/1 o Y/N de forma consistente.		No aplica

23	En los algoritmos u operaciones matemáticas se ha controlado que las variables contengan ceros o negativos.		No aplica
24	Luego de ser utilizados, se limpian / destruyen las variables, objetos creados.		No aplica
25	Se siguen los estándares de desarrollo definidos.	0	OK
26	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica

<b>Otros definidos por el proyecto - (Insertar filas adicionales si es necesario)</b>			
27			
28			

### Prueba Funcional

En la siguiente sección, mencionar las pruebas relacionadas con la funcionalidad programada. Indicar en la columna “Caso de Uso / Especificación Funcional”.

#	Caso de Uso / Especificación Funcional	Especificación del documento Nombre y Versión	Número de errores detectados	Estado
29	Caso de Uso de Feligrés		0	Aprobado
30				
31				

Total de Errores Detectados	<b>0</b>
-----------------------------	----------

Tabla 35 Pruebas Unitarias del Módulo de Feligrés

### Formato de Pruebas Unitarias

<b>Módulo</b>	Sacramentos y Reportes de Sacramentos
<b>Desarrollador</b>	Juan Castillo
<b>Fecha</b>	26/08/2013

#	Descripción de la Prueba	Número de errores detectados	Estado
---	--------------------------	------------------------------	--------

Nota: Si no se detectan errores, colocar '0'

<b>Interfaz</b>			
1	El diseño de la pantalla / página, creada o modificada sigue el estándar definido en el proyecto	0	OK
2	Al hacer resize, maximize, etc., no se altera la apariencia de la pantalla.		No aplica
3	Los campos en pantalla se habilitan y deshabilitan correctamente, según la transacción o estado de la pantalla.	0	OK
4	La longitud de los campos en pantalla corresponde a la de los campos en base de datos.	0	OK
5	Los tipos de dato de los campos en pantalla corresponden a los definidos en base de datos.	0	OK
6	Los datos se ingresan, muestran y almacenan en mayúsculas o minúsculas, según lo definido.	0	OK
7	La validación de campos mandatorios funciona correctamente para todos los casos.	0	OK
8	Los campos numéricos soportan decimales, negativos, o ceros según lo definido.	0	OK
9	Los campos de fecha validan correctamente el formato definido y el rango soportado por el lenguaje de programación y por el motor de base de datos.	0	OK
10	El desplazamiento por los componentes de pantalla con la tecla TAB es ordenado.	0	OK

11	Al cancelar o grabar una transacción, los campos en pantalla se inicializan correctamente.	0	OK
12	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica
<b>Reportes</b>			
13	La longitud de los campos de salida del reporte soporta el tamaño de todos los datos definidos.	0	OK
14	El reporte se puede generar correctamente en todos los formatos definidos (Excel, texto, pdf, etc.).	0	OK
15	El formato de reporte mantiene el estándar definido por el proyecto.	0	OK
16	El reporte se puede generar correctamente con la amplitud máxima de los filtros(es decir procesando la mayor cantidad de información que permitan los filtros mandatorios.)	0	OK
17	El reporte se puede generar correctamente ingresando un valor en todos los filtros incluso los no mandatorios.		No aplica
<b>Base de Datos</b>			
18	Los objetos nuevos (SP, Tablas, Funciones, etc.) siguen los estándares de nomenclatura definidos para el proyecto.	0	OK
19	Las consultas nuevas o modificadas han sido probadas con un volumen de información similar al del ambiente de producción. En caso de ser así, ¿se comportan según lo esperado para el proyecto (en tiempo y recursos)?		No aplica
20	Se controla o tolera si el resultado de la consulta a base de datos es vacío(aun cuando la tabla debería contener información en condiciones normales)		No aplica
<b>Otros</b>			
21	El código compila sin warnings asociados al módulo u opción probada.	0	OK
22	Los flags utilizados corresponden a 0/1 o Y/N de forma consistente.		No aplica

23	En los algoritmos u operaciones matemáticas se ha controlado que las variables contengan ceros o negativos.	0	OK
24	Luego de ser utilizados, se limpian / destruyen las variables, objetos creados.		No aplica
25	Se siguen los estándares de desarrollo definidos.	0	OK
26	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica

**Otros definidos por el proyecto - (Insertar filas adicionales si es necesario)**

27			
28			

**Prueba Funcional**

En la siguiente sección, mencionar las pruebas relacionadas con la funcionalidad programada. Indicar en la columna “Caso de Uso / Especificación Funcional”.

#	Caso de Uso / Especificación Funcional	Especificación del documento Nombre y Versión	Número de errores detectados	Estado
29	Caso de Uso de Sacramentos		0	Aprobado
30	Caso de Uso del Libro Sacramental		0	Aprobado
31	Caso de Uso de Templos		0	Aprobado

Total de Errores Detectados	0
-----------------------------	---

Tabla 36 Pruebas Unitarias del Módulo de Sacramentos y Reportes de Sacramentos

### Formato de Pruebas Unitarias

<b>Módulo</b>	Personal
<b>Desarrollador</b>	Juan Castillo
<b>Fecha</b>	19/09/2013

#	Descripción de la Prueba	Número de errores detectados	Estado
---	--------------------------	------------------------------	--------

**Nota:** Si no se detectan errores, colocar '0'

<b>Interfaz</b>			
1	El diseño de la pantalla / página, <u>creada o modificada</u> sigue el estándar definido en el proyecto	0	OK
2	Al hacer resize, maximize, etc., no se altera la apariencia de la pantalla.		No aplica
3	Los campos en pantalla se habilitan y deshabilitan correctamente, según la transacción o estado de la pantalla.	0	OK
4	La longitud de los campos en pantalla corresponde a la de los campos en base de datos.	0	OK
5	Los tipos de dato de los campos en pantalla corresponden a los definidos en base de datos.	0	OK
6	Los datos se ingresan, muestran y almacenan en mayúsculas o minúsculas, según lo definido.	0	OK
7	La validación de campos mandatorios funciona correctamente para todos los casos.	0	OK
8	Los campos numéricos soportan decimales, negativos, o ceros según lo definido.	0	OK
9	Los campos de fecha validan correctamente el formato definido y el rango soportado por el lenguaje de programación y por el motor de base de datos.	0	OK
10	El desplazamiento por los componentes de pantalla con la tecla TAB es ordenado.	0	OK

11	Al cancelar o grabar una transacción, los campos en pantalla se inicializan correctamente.	0	OK
12	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica
<b>Reportes</b>			
13	La longitud de los campos de salida del reporte soporta el tamaño de todos los datos definidos.	0	OK
14	El reporte se puede generar correctamente en todos los formatos definidos (Excel, texto, pdf, etc.).	0	OK
15	El formato de reporte mantiene el estándar definido por el proyecto.	0	OK
16	El reporte se puede generar correctamente con la amplitud máxima de los filtros(es decir procesando la mayor cantidad de información que permitan los filtros mandatorios.)	0	OK
17	El reporte se puede generar correctamente ingresando un valor en todos los filtros incluso los no mandatorios.		No aplica
<b>Base de Datos</b>			
18	Los objetos nuevos (SP, Tablas, Funciones, etc.) siguen los estándares de nomenclatura definidos para el proyecto.	0	OK
19	Las consultas nuevas o modificadas han sido probadas con un volumen de información similar al del ambiente de producción. En caso de ser así, ¿se comportan según lo esperado para el proyecto (en tiempo y recursos)?		No aplica
20	Se controla o tolera si el resultado de la consulta a base de datos es vacío(aun cuando la tabla debería contener información en condiciones normales)		No aplica
<b>Otros</b>			
21	El código compila sin warnings asociados al módulo u opción probada.	0	OK
22	Los flags utilizados corresponden a 0/1 o Y/N de forma consistente.		No aplica

23	En los algoritmos u operaciones matemáticas se ha controlado que las variables contengan ceros o negativos.	0	OK
24	Luego de ser utilizados, se limpian / destruyen las variables, objetos creados.		No aplica
25	Se siguen los estándares de desarrollo definidos.	0	OK
26	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica

**Otros definidos por el proyecto - (Insertar filas adicionales si es necesario)**

27			
28			

**Prueba Funcional**

*En la siguiente sección, mencionar las pruebas relacionadas con la funcionalidad programada. Indicar en la columna “Caso de Uso / Especificación Funcional”.*

#	Caso de Uso / Especificación Funcional	Especificación del documento Nombre y Versión	Número de errores detectados	Estado
29	Caso de Uso de Personal		0	Aprobado
30				
31				

Total de Errores Detectados	0
-----------------------------	---

Tabla 37 Pruebas Unitarias del Módulo de Personal

### Formato de Pruebas Unitarias

<b>Modulo</b>	Comunidades
<b>Desarrollador</b>	Juan Castillo
<b>Fecha</b>	02/10/2013

#	Descripción de la Prueba	Número de Errores Detectados	Estado
---	--------------------------	------------------------------	--------

Nota: Si no se detectan errores, colocar '0'

<b>Interfaz</b>			
1	El diseño de la pantalla / página, <u>creada o modificada</u> sigue el estándar definido en el proyecto	0	OK
2	Al hacer resize, maximize, etc., no se altera la apariencia de la pantalla.		No aplica
3	Los campos en pantalla se habilitan y deshabilitan correctamente, según la transacción o estado de la pantalla.	0	OK
4	La longitud de los campos en pantalla corresponde a la de los campos en base de datos.	0	OK
5	Los tipos de dato de los campos en pantalla corresponden a los definidos en base de datos.	0	OK
6	Los datos se ingresan, muestran y almacenan en mayúsculas o minúsculas, según lo definido.	0	OK
7	La validación de campos mandatorios funciona correctamente para todos los casos.	0	OK
8	Los campos numéricos soportan decimales, negativos, o ceros según lo definido.	0	OK
9	Los campos de fecha validan correctamente el formato definido y el rango soportado por el lenguaje de programación y por el motor de base de datos.	0	OK
10	El desplazamiento por los componentes de pantalla con la tecla TAB es ordenado.	0	OK

11	Al cancelar o grabar una transacción, los campos en pantalla se inicializan correctamente.	0	OK
12	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica
<b>Reportes</b>			
13	La longitud de los campos de salida del reporte soporta el tamaño de todos los datos definidos.	0	OK
14	El reporte se puede generar correctamente en todos los formatos definidos (Excel, texto, pdf, etc.).	0	OK
15	El formato de reporte mantiene el estándar definido por el proyecto.	0	OK
16	El reporte se puede generar correctamente con la amplitud máxima de los filtros(es decir procesando la mayor cantidad de información que permitan los filtros mandatorios.)	0	OK
17	El reporte se puede generar correctamente ingresando un valor en todos los filtros incluso los no mandatorios.		No aplica
<b>Base de Datos</b>			
18	Los objetos nuevos (SP, Tablas, Funciones, etc.) siguen los estándares de nomenclatura definidos para el proyecto.	0	OK
19	Las consultas nuevas o modificadas han sido probadas con un volumen de información similar al del ambiente de producción. En caso de ser así, ¿se comportan según lo esperado para el proyecto (en tiempo y recursos)?		No aplica
20	Se controla o tolera si el resultado de la consulta a base de datos es vacío(aun cuando la tabla debería contener información en condiciones normales)		No aplica
<b>Otros</b>			
21	El código compila sin warnings asociados al módulo u opción probada.	0	OK
22	Los flags utilizados corresponden a 0/1 o Y/N de forma consistente.		No aplica

23	En los algoritmos u operaciones matemáticas se ha controlado que las variables contengan ceros o negativos.	0	OK
24	Luego de ser utilizados, se limpian / destruyen las variables, objetos creados.		No aplica
25	Se siguen los estándares de desarrollo definidos.	0	OK
26	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica

**Otros definidos por el proyecto - (Insertar filas adicionales si es necesario)**

27			
28			

**Prueba Funcional**

*En la siguiente sección, mencionar las pruebas relacionadas con la funcionalidad programada. Indicar en la columna “Caso de Uso / Especificación Funcional”.*

#	Caso de Uso / Especificación Funcional	Especificación del documento Nombre y Versión	Número de errores detectados	Estado
29	Caso de Uso de Comunidades		0	Aprobado
30	Caso de Uso de Asignación		0	Aprobado
31				

Total de Errores Detectados	0
-----------------------------	---

Tabla 38 Pruebas Unitarias del Módulo de Comunidades

### Formato de Pruebas Unitarias

<b>Módulo</b>	Reportes de Asignación y Backup's
<b>Desarrollador</b>	Juan Castillo
<b>Fecha</b>	14/10/2013

#	Descripción de la Prueba	Número de Errores Detectados	Estado
---	--------------------------	------------------------------	--------

Nota: Si no se detectan errores, colocar '0'

<i>Interfaz</i>			
1	El diseño de la pantalla / página, <u>creada o modificada</u> sigue el estándar definido en el proyecto	0	OK
2	Al hacer resize, maximize, etc., no se altera la apariencia de la pantalla.		No aplica
3	Los campos en pantalla se habilitan y deshabilitan correctamente, según la transacción o estado de la pantalla.	0	OK
4	La longitud de los campos en pantalla corresponde a la de los campos en base de datos.	0	OK
5	Los tipos de dato de los campos en pantalla corresponden a los definidos en base de datos.	0	OK
6	Los datos se ingresan, muestran y almacenan en mayúsculas o minúsculas, según lo definido.	0	OK
7	La validación de campos mandatorios funciona correctamente para todos los casos.	0	OK
8	Los campos numéricos soportan decimales, negativos, o ceros según lo definido.	0	OK
9	Los campos de fecha validan correctamente el formato definido y el rango soportado por el lenguaje de programación y por el motor de base de datos.	0	OK
10	El desplazamiento por los componentes de pantalla con la tecla TAB es ordenado.	0	OK

11	Al cancelar o grabar una transacción, los campos en pantalla se inicializan correctamente.	0	OK
12	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica
<b>Reportes</b>			
13	La longitud de los campos de salida del reporte soporta el tamaño de todos los datos definidos.	0	OK
14	El reporte se puede generar correctamente en todos los formatos definidos (Excel, texto, pdf, etc.).	0	OK
15	El formato de reporte mantiene el estándar definido por el proyecto.	0	OK
16	El reporte se puede generar correctamente con la amplitud máxima de los filtros(es decir procesando la mayor cantidad de información que permitan los filtros mandatorios.)	0	OK
17	El reporte se puede generar correctamente ingresando un valor en todos los filtros incluso los no mandatorios.		No aplica
<b>Base de Datos</b>			
18	Los objetos nuevos (SP, Tablas, Funciones, etc.) siguen los estándares de nomenclatura definidos para el proyecto.	0	OK
19	Las consultas nuevas o modificadas han sido probadas con un volumen de información similar al del ambiente de producción. En caso de ser así, ¿se comportan según lo esperado para el proyecto (en tiempo y recursos)?		No aplica
20	Se controla o tolera si el resultado de la consulta a base de datos es vacío(aun cuando la tabla debería contener información en condiciones normales)		No aplica
<b>Otros</b>			
21	El código compila sin warnings asociados al módulo u opción probada.	0	OK
22	Los flags utilizados corresponden a 0/1 o Y/N de forma consistente.		No aplica

23	En los algoritmos u operaciones matemáticas se ha controlado que las variables contengan ceros o negativos.	0	OK
24	Luego de ser utilizados, se limpian / destruyen las variables, objetos creados.		No aplica
25	Se siguen los estándares de desarrollo definidos.	0	OK
26	La navegabilidad entre pantallas funciona de acuerdo a lo especificado (si se trata de una aplicación web).		No aplica

**Otros definidos por el proyecto - (Insertar filas adicionales si es necesario)**

27			
28			

**Prueba Funcional**

*En la siguiente sección, mencionar las pruebas relacionadas con la funcionalidad programada. Indicar en la columna “Caso de Uso / Especificación Funcional”.*

#	Caso de Uso / Especificación Funcional	Especificación del documento Nombre y Versión	Número de errores detectados	Estado
29				
30				
31				

Total de Errores Detectados	0
-----------------------------	---

Tabla 39 Pruebas Unitarias del Módulo de Reportes de Asignación y Backup's

#### 4.3.3. Medición del valor ganado

##### Control de Avance

EDT	Actividad	ABRIL	MAYO	JUNIO	JULIO	AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE
		MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8
1.1.1	Modelamiento de Negocio	47.00%	53.00%						
1.1.2	Requisitos		100.00%						
1.2.1	Análisis de Documentos		100.00%						
1.2.2	Análisis de Requisitos		100.00%						
1.2.3	Diseño de Diagramas		9.00%	91.00%					
1.2.4	Modelar Base de Datos			100.00%					
1.2.5	Diseñar Prototipo de Interfaces		52.00%	48.00%					
1.3.1	Implementar Base de Datos en MySql			100.00%					
1.3.2	Diseñar Interfaces del Sistema			100.00%					
1.3.3	Sub-Sistema de Control de Sacramentos			49.00%	35.00%	16.00%			
1.3.4	Sub-Sistema de Asignación de Personal				48.00%	0.00%			
1.3.5	Codificar Módulo de Backups					0.00%			
1.3.6	Ejecutar Pruebas del Módulo					0.00%			
1.3.7	Elaborar Acta de Pruebas					0.00%			
1.3.8	Elaborar Manual Técnico					0.00%			
1.3.9	Elaborar Manual de Usuario					0.00%	0.00%		
1.4.1	Ejecutar Pruebas Integrales						0.00%		

1.4.2	Corrección de Errores Detectados		0.00%
1.4.3	Elaborar Acta de Pruebas		0.00%
1.4.4	Implantación del Sistema en la Parroquia		0.00%
1.4.5	Capacitación a los Usuarios		0.00%
1.4.6	Entrega de Manuales		0.00%
1.4.7	Elaborar Acta de Aceptación		0.00%
1.5.1.1	Elaborar Acta de Constitución	65.00%      35.00%	
1.5.1.2	Identificar Interesados	100.00%	
1.5.2.1	Elaborar Plan de Gestión de Py	100.00%	
1.5.2.2	Elaborar Plan de Gestión de Cambios	100.00%	
1.5.2.3	Elaborar Plan de Gestión de Requerimientos	100.00%	
1.5.2.4	Elaborar Plan de Gestión de Alcance	100.00%	
1.5.2.5	Elaborar EDT	100.00%	
1.5.2.6	Elaborar Cronograma	100.00%	
1.5.2.7	Elaborar Presupuesto	100.00%	
1.5.2.8	Elaborar Plan de Gestión de Calidad del Py	100.00%	
1.5.2.9	Elaborar Plan de Gestión de Comunicaciones	100.00%	
1.5.2.10	Elaborar Plan de Gestión de Riesgos	100.00%      100.00% %	
1.5.2.11	Elaborar Plan de Gestión de	100.00%	

	RRHH	%					
1.5.2.12	Elaborar Plan de Gestión de Adquisiciones	100.00 %					
1.5.3.1	Dirigir y Gestionar la ejecución del Py	100.00 %					
1.5.3.2	Aseguramiento de la Calidad	100.00 %					
1.5.3.3	Definir Acciones Correctivas recomendadas	100.00 %					
1.5.3.4	Realizar la distribución de la información	100.00 %					
1.5.4.1	Supervisar y controlar el trabajo del Py	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.2	Control Integrado de Cambios	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.3	Verificación del Alcance	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.4	Control del Alcance	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.5	Control del Cronograma	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.6	Control de Costos	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.7	Control y Seguimiento de Riesgos	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.4.8	Control de Calidad	12.50%	12.50%	12.50%	12.50%	12.50%	7.20% 0.00% 0.00%
1.5.5.1.1	Lecciones Aprendidas	0.00%					
1.5.5.2.1	Aceptación Final del Py	0.00%					

Tabla 40 Cuadro de Control de Avance

Costo Real

EDT	Actividad	TOTAL	ABRIL	MAYO	JUNIO	JULIO	AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE
			MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8
1.1.1	Modelamiento de Negocio	S/. 182.00	S/. 85.54	S/. 96.46						
1.1.2	Requisitos	S/. 38.00		S/. 38.00						
1.2.1	Análisis de Documentos	S/. 36.00		S/. 36.00						
1.2.2	Análisis de Requisitos	S/. 36.00		S/. 36.00						
1.2.3	Diseño de Diagramas	S/. 192.00		S/. 17.28	174.72					
1.2.4	Modelar Base de Datos	S/. 96.00		S/. 96.00						
1.2.5	Diseñar Prototipo de Interfaces	S/. 437.00		S/. 227.24	S/. 209.76					
1.3.1	Implementar Base de Datos en MySql	S/. 362.00		S/. 362.00						
1.3.2	Diseñar Interfaces del Sistema	S/. 769.00		S/. 769.00						
1.3.3	Sub-Sistema de Control de Sacramentos	S/. 3,036.00		S/. 1,487.64	S/. 1,062.60	S/. 485.76				
1.3.4	Sub-Sistema de Asignación de Personal	S/. 1,066.65					S/. 787.68	S/. 0.00		
1.3.5	Codificar Módulo de Backups	S/. 0.00					S/. 0.00			
1.3.6	Ejecutar Pruebas del Módulo	S/. 0.00					S/. 0.00			
1.3.7	Elaborar Acta de Pruebas	S/. 0.00					S/. 0.00			
1.3.8	Elaborar Manual Técnico	S/. 0.00					S/. 0.00			
1.3.9	Elaborar Manual de Usuario	S/. 0.00					S/. 0.00	S/. 0.00		

1.4.1	Ejecutar Pruebas Integrales	S/. 0.00		S/. 0.00
1.4.2	Corrección de Errores Detectados	S/. 0.00		S/. 0.00
1.4.3	Elaborar Acta de Pruebas	S/. 0.00		S/. 0.00
1.4.4	Implantación del Sistema en la Parroquia	S/. 0.00		S/. 0.00
1.4.5	Capacitación a los Usuarios	S/. 0.00		S/. 0.00
1.4.6	Entrega de Manuales	S/. 0.00		S/. 0.00
1.4.7	Elaborar Acta de Aceptación	S/. 0.00		S/. 0.00
1.5.1.1	Elaborar Acta de Constitución	S/. 112.00	S/. 72.00	S/. 40.00
1.5.1.2	Identificar Interesados	S/. 12.00	S/. 12.00	
1.5.2.1	Elaborar Plan de Gestión de Py	S/. 80.00		S/. 80.00
1.5.2.2	Elaborar Plan de Gestión de Cambios	S/. 80.00		S/. 80.00
1.5.2.3	Elaborar Plan de Gestión de Requerimientos	S/. 48.00		S/. 48.00
1.5.2.4	Elaborar Plan de Gestión de Alcance	S/. 128.00		S/. 128.00
1.5.2.5	Elaborar EDT	S/. 24.00		S/. 24.00
1.5.2.6	Elaborar Cronograma	S/. 48.00		S/. 48.00
1.5.2.7	Elaborar Presupuesto	S/. 80.00		S/. 80.00
1.5.2.8	Elaborar Plan de Gestión de Calidad del Py	S/. 128.00		S/. 128.00

1.5.2.9	Elaborar Plan de Gestión de Comunicaciones	S/. 128.00			S/. 128.00						
1.5.2.10	Elaborar Plan de Gestión de Riesgos	S/. 160.00			S/. 80.00	S/. 80.00					
1.5.2.11	Elaborar Plan de Gestión de RRHH	S/. 240.00					S/. 240.00				
1.5.2.12	Elaborar Plan de Gestión de Adquisiciones	S/. 176.00					S/. 176.00				
1.5.3.1	Dirigir y Gestionar la ejecución del Py	S/. 176.00			S/. 176.00						
1.5.3.2	Aseguramiento de la Calidad	S/. 208.00			S/. 208.00						
1.5.3.3	Definir Acciones Correctivas recomendadas	S/. 128.00			S/. 128.00						
1.5.3.4	Realizar la distribución de la información	S/. 128.00			S/. 128.00						
1.5.4.1	Supervisar y controlar el trabajo del Py	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.2	Control Integrado de Cambios	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.3	Verificación del Alcance	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.4	Control del Alcance	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.5	Control del Cronograma	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.6	Control de Costos	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.7	Control y Seguimiento de Riesgos	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	
1.5.4.8	Control de Calidad	S/. 102.90	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 19.00	S/. 10.94	S/. 0.00	S/. 0.00	

1.5.5.1.1	Lecciones Aprendidas	S/. 0.00							S/. 0.00
1.5.5.2.1	Aceptación Final del Py	S/. 0.00							S/. 0.00
<b>Total del Costo Mensual del Proyecto</b>		<b>S/. 321.54</b>	<b>S/. 575.74</b>	<b>S/. 1,313.96</b>	<b>S/. 4,116.40</b>	<b>S/. 1,214.60</b>	<b>S/. 1,360.99</b>	<b>S/. 0.00</b>	<b>S/. 0.00</b>
<b>Total de Costo del Proyecto</b>		<b>S/. 321.54</b>	<b>S/. 897.28</b>	<b>S/. 2,211.24</b>	<b>S/. 6,327.64</b>	<b>S/. 7,542.24</b>	<b>S/. 8,903.23</b>	<b>S/. 0.00</b>	<b>S/. 0.00</b>

Tabla 41 Cuadro de Costo Real

### Resumen

	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8
Presupuesto	S/. 322.00	S/. 898.00	S/. 2,141.00	S/. 6,254.00	S/. 7,493.00	S/. 9,002.50	S/. 10,621.00	S/. 11,878.00
Valor Ganado	S/. 321.54	S/. 897.28	S/. 2,211.24	S/. 6,327.64	S/. 7,542.24	S/. 8,903.23	S/. 0.00	S/. 0.00
Costo Real	S/. 322.00	S/. 898.00	S/. 2,141.00	S/. 6,254.00	S/. 7,493.00	S/. 8,903.23	S/. 0.00	S/. 0.00

Tabla 42 Tabla Resumen de Presupuesto, Valor Ganado y Costo Real

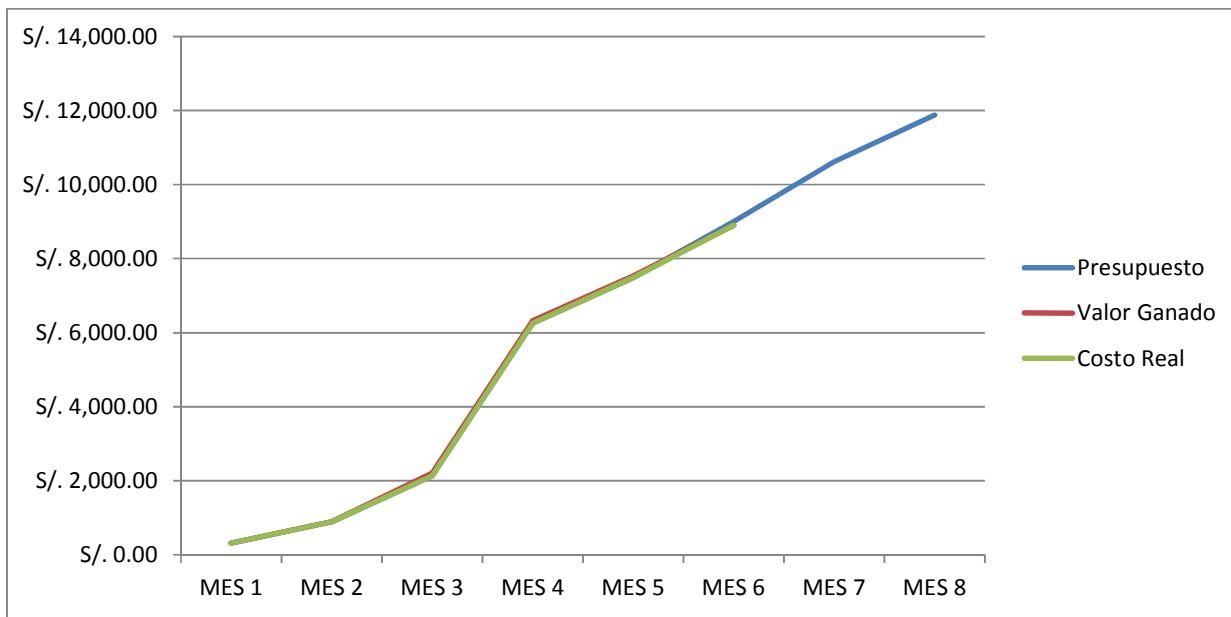


Ilustración 53 Gráfico de Valor Ganado

#### 4.3.4. Métricas y evaluación del desempeño

Periodo 03-07-2013 al 31-07-2013

ESTADO DE AVANCE DE LOS ENTREGABLES							
FASE	ENTREGABLE	ESTADO DE AVANCE				OBSERVACIONES	
Construcción	Módulo de Feligrés	Se completó con el desarrollo del primer módulo. Las pruebas arrojaron falencias en ciertos campos, por lo que se encuentra en correcciones.					
	Módulo de Sacramentos	A la fecha, el desarrollo del módulo se encuentra en un 45 % aproximadamente.					
ACTIVIDADES INICIADAS EN EL PERÍODO							
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO			REAL		
		FECHA INICIO	FECHA FIN	DURACIÓN	RECURSO HUMANO	FECHA INICIO	
Implementar Base de Datos en el gestor	Implementar Base de Datos en MySql	03-07-2013	05-07-2013	2 días	-Administrador de Base de Datos	03-07-2013	-Administrador de Base de Datos
Interfaces del Sistema	Diseñar interfaces del sistema	05-07-2013	10-07-2013	3 días	-Jefe de Proyecto -Programador Senior -Programador Junior	05-07-2013	-Jefe de Proyecto -Programador Senior -Programador Junior
Sub-Sistema Control de Sacramentos	Codificar Módulo de Feligrés	10-07-2013	17-07-2013	1 semana	-Programador Senior -Programador Junior	10-07-2013	-Programador Senior -Programador Junior
	Ejecutar Pruebas del Módulo	17-07-2013	18-07-2013	1 día	-Analista de Calidad	17-07-2013	-Analista de Calidad
	Elaborar Acta de Pruebas	18-07-2013	19-07-2013	1 día	-Analista de Calidad	18-07-2013	-Analista de Calidad
	Codificar Módulo	19-07-2013	19-08-2013	3 semanas	-Programador Senior	19-07-2013	-Programador Senior

	de Sacramen tos				-Programador Junior		-Programador Junior	
<b>ACTIVIDADES FINALIZADAS EN EL PERÍODO</b>								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	FECHA INICIO	FECHA FIN	RECURSOS	
Implementar Base de Datos en el gestor	Implementar Base de Datos en MySql	03-07-2013	05-07-2013	2 días	-Administrador de Base de Datos	03-07-2013	-Administrador de Base de Datos	
Interfaces del Sistema	Diseñar interfaces del sistema	05-07-2013	10-07-2013	3 días	-Jefe de Proyecto -Programador Senior -Programador Junior	05-07-2013	-Jefe de Proyecto -Programador Senior -Programador Junior	
Sub-Sistema Control de Sacramentos	Codificar Módulo de Feligrés	10-07-2013	17-07-2013	1 semana	-Programador Senior -Programador Junior	10-07-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	17-07-2013	18-07-2013	1 día	-Analista de Calidad	17-07-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	18-07-2013	19-07-2013	1 día	-Analista de Calidad	18-07-2013	-Analista de Calidad	
<b>MÉTRICAS DE CALIDAD EN EL PERÍODO</b>								
FACTOR RELEVANTE DE CALIDAD	OBJETIVO DE CALIDAD			MEDICIÓN DE MÉTRICA		OBSERVACIONES		
				FECHA	RESULTADO OBTENIDO			
Acta de Prueba	Obtener la aprobación de todos los elementos del módulo			19-07-2013	Desaprobatorio	Uno de los elementos no pudo pasar la aprobación del cliente, por lo que se encuentra en proceso de corrección de errores.		
<b>CONTROLES DE CALIDAD EN EL PERÍODO</b>								
ENTREGABLE	ESTÁNDAR DE CALIDAD			OBSERVACIONES		OBSERVACIONES		

		APLICABLE	FECHA	RESULTADO OBTENIDO	
Módulo de Feligrés	Revisión de funciones y fechas de entrega		19-07-2013	Desaprobatorio	Se pasó por alto algunos campos al momento de registrar datos.
<b>ACTIVIDADES EN PROCESO A LA FECHA</b>					
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	FECHA DE FIN PROGRAMADA	FECHA DE FIN ESTIMADA	% DE AVANCE A LA FECHA	OBSERVACIONES
Sub-Sistema Control de Sacramentos	Codificar Módulo de Sacramentos	19-08-2013	19-08-2013	45%	

Tabla 43 Estado de Avance del Proyecto N° 1

Periodo del 01-08-2013 al 30-08-2013

ESTADO DE AVANCE DE LOS ENTREGABLES								
FASE	ENTREGABLE	ESTADO DE AVANCE						OBSERVACIONES
Construcción	Módulo de Sacramentos	Se completó con el desarrollo del segundo módulo. Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.						
	Módulo de Reportes de Sacramentos	Se completó con el desarrollo del tercer módulo. Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.						
ACTIVIDADES INICIADAS EN EL PERÍODO								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	RECURSO HUMANO	FECHA INICIO	RECURSOS	
Sub-Sistema Control de Sacramentos	Codificar Módulo de Sacramentos	19-07-2013	19-08-2013	3 semanas	-Programador Senior -Programador Junior	19-07-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	19-08-2013	27-08-2013	3 días	-Analista de Calidad	19-08-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	27-08-2013	29-08-2013	1 día	-Analista de Calidad	28-08-2013	-Analista de Calidad	Se entregó conjuntamente con el módulo

								Reportes de Sacramentos
	Codificar Módulo de Reportes de Sacramentos	29-08-2013	06-09-2013	2 días	-Programador Senior -Programador Junior	22-08-2013	-Programador Senior -Programador Junior	Adelanto de fechas
	Ejecutar Pruebas del Módulo	06-09-2013	09-09-2013	1 día	-Analista de Calidad	26-08-2013	-Analista de Calidad	Adelanto de fechas
	Elaborar Acta de Pruebas	09-09-2013	11-09-2013	3 horas	-Analista de Calidad	28-08-2013	-Analista de Calidad	Adelanto de fechas
<b>ACTIVIDADES FINALIZADAS EN EL PERÍODO</b>								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	FECHA INICIO	FECHA FIN	RECURSOS	
Sub-Sistema Control de Sacramentos	Codificar Módulo de Sacramentos	19-07-2013	19-08-2013	3 semanas	-Programador Senior -Programador Junior	19-07-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	19-08-2013	27-08-2013	3 días	-Analista de Calidad	19-08-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	27-08-2013	29-08-2013	1 día	-Analista de Calidad	27-08-2013	-Analista de Calidad	Se entregó conjuntamente con el módulo Reportes de Sacramentos
	Codificar Módulo de Reportes de Sacramentos	29-08-2013	06-09-2013	2 días	-Programador Senior -Programador Junior	22-08-2013	-Programador Senior -Programador Junior	Adelanto de fechas
	Ejecutar Pruebas del Módulo	06-09-2013	09-09-2013	1 día	-Analista de Calidad	26-08-2013	-Analista de Calidad	Adelanto de fechas

	Elaborar Acta de Pruebas	09-09-2013	11-09-2013	3 horas	-Analista de Calidad	28-08-2013	-Analista de Calidad	Adelanto de fechas				
<b>MÉTRICAS DE CALIDAD EN EL PERÍODO</b>												
FACTOR RELEVANTE DE CALIDAD	OBJETIVO DE CALIDAD	MEDICIÓN DE MÉTRICA			OBSERVACIONES							
		FECHA	RESULTADO OBTENIDO									
Acta de Prueba	Obtener la aprobación de todos los elementos del módulo	28-08-2013	Aprobado	No hubo ningún fallo al momento de ejecutar las pruebas de ambos módulos.								
<b>CONTROLES DE CALIDAD EN EL PERÍODO</b>												
ENTREGABLE	ESTÁNDAR DE CALIDAD APLICABLE	OBSERVACIONES			OBSERVACIONES							
		FECHA	RESULTADO OBTENIDO									
Módulo de Sacramentos y Reportes de Sacramentos	Revisión de funciones y fechas de entrega	28-08-2013	Aprobado									
<b>ACTIVIDADES EN PROCESO A LA FECHA</b>												
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	FECHA DE FIN PROGRAMADA	FECHA DE FIN ESTIMADA	% DE AVANCE A LA FECHA	OBSERVACIONES							

Tabla 44 Estado de Avance del Proyecto Nº 2

Periodo del 31-08-2013 al 27-09-2013

ESTADO DE AVANCE DE LOS ENTREGABLES			
FASE	ENTREGABLE	ESTADO DE AVANCE	OBSERVACIONES
Construcción	Módulo de Personal	Se completó con el desarrollo del cuarto módulo. Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	
	Módulo de Comunidades	Se sigue con el desarrollo del módulo. A la fecha se lleva un avance del 95% aproximadamente.	

ACTIVIDADES INICIADAS EN EL PERÍODO								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	RECURSO HUMANO	FECHA INICIO	RECURSOS	
Sub-Sistema Control de Sacramentos	Codificar Módulo de Personal	11-09-2013	19-09-2013	3 días	-Programador Senior -Programador Junior	11-09-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	19-09-2013	20-09-2013	1 día	-Analista de Calidad	19-09-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	23-09-2013	23-09-2013	3 horas	-Analista de Calidad	23-09-2013	-Analista de Calidad	
	Codificar Módulo de Comunidades	23-09-2013	30-09-2013	2.5 días	-Programador Senior -Programador Junior	23-09-2013	-Programador Senior -Programador Junior	
ACTIVIDADES FINALIZADAS EN EL PERÍODO								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	FECHA INICIO	FECHA FIN	RECURSOS	
Sub-Sistema Asignación de Personal	Codificar Módulo de Personal	11-09-2013	19-09-2013	3 días	-Programador Senior -Programador Junior	11-09-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	19-09-2013	20-09-2013	1 día	-Analista de Calidad	19-09-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	23-09-2013	23-09-2013	3 horas	-Analista de Calidad	23-09-2013	-Analista de Calidad	
	Codificar Módulo de Comunidades	23-09-2013	30-09-2013	2.5 días	-Programador Senior -Programador Junior	23-09-2013	-Programador Senior -Programador Junior	

MÉTRICAS DE CALIDAD EN EL PERÍODO					
FACTOR RELEVANTE DE CALIDAD	OBJETIVO DE CALIDAD	MEDICIÓN DE MÉTRICA		OBSERVACIONES	
		FECHA	RESULTADO OBTENIDO		
Acta de Prueba	Obtener la aprobación de todos los elementos del módulo	23-09-2013	Aprobado	No hubo ningún fallo al momento de ejecutar las pruebas de ambos módulos.	
CONTROLES DE CALIDAD EN EL PERÍODO					
ENTREGABLE	ESTÁNDAR DE CALIDAD APLICABLE	OBSERVACIONES		OBSERVACIONES	
		FECHA	RESULTADO OBTENIDO		
Módulo de Personal	Revisión de funciones y fechas de entrega	23-09-2013	Aprobado		
ACTIVIDADES EN PROCESO A LA FECHA					
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	FECHA DE FIN PROGRAMADA	FECHA DE FIN ESTIMADA	% DE AVANCE A LA FECHA	OBSERVACIONES
Sub-Sistema Asignación de Personal	Codificar Módulo de Comunidades	30-09-2013	30-09-2013	95%	

Tabla 45 Estado de Avance del Proyecto N° 3

Periodo del 27-09-2013 al 25-10-2013

ESTADO DE AVANCE DE LOS ENTREGABLES			
FASE	ENTREGABLE	ESTADO DE AVANCE	OBSERVACIONES
Construcción	Módulo de Comunidades	Se completó con el desarrollo del quinto módulo. Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	
	Módulo de Reportes de Asignación	Se completó con el desarrollo del sexto módulo. Las pruebas fueron satisfactorias, por lo que se sigue con el avance del sistema.	
	Módulo de Backups	Se completó con el desarrollo del séptimo módulo. Las pruebas fueron satisfactorias.	

ACTIVIDADES INICIADAS EN EL PERÍODO								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	RECURSO HUMANO	FECHA INICIO	RECURSOS	
Sub-Sistema Control de Sacramentos	Codificar Módulo de Comunidades	23-09-2013	30-09-2013	2.5 días	-Programador Senior -Programador Junior	23-09-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	02-10-2013	03-10-2013	5 horas	-Analista de Calidad	02-10-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	03-10-2013	04-10-2013	4 horas	-Analista de Calidad	03-10-2013	-Analista de Calidad	
	Codificar Módulo de Reportes de Asignación	04-10-2013	14-10-2013	3.5 días	-Programador Senior -Programador Junior	04-10-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	14-10-2013	16-10-2013	2 horas	-Analista de Calidad	12-10-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	16-10-2013	16-10-2013	2 horas	-Analista de Calidad	14-10-2013	-Analista de Calidad	
Módulo de Backups	Codificar Módulo de Backups	16-10-2013	23-10-2013	2.5 días	-Programador Senior -Programador Junior	10-10-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	24-10-2013	24-10-2013	3 horas	-Analista de Calidad	13-10-2013	-Analista de Calidad	

	Elaborar Acta de Pruebas	24-10-2013	25-10-2013	2 horas	-Analista de Calidad	14-10-2013	-Analista de Calidad	
<b>ACTIVIDADES FINALIZADAS EN EL PERÍODO</b>								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL	OBSERVACIONES	
Sub-Sistema Asignación de Personal	Codificar Módulo de Comunidades	23-09-2013	30-09-2013	2.5 días	-Programador Senior -Programador Junior	30-09-2013	-Programador Senior -Programador Junior	
	Ejecutar Pruebas del Módulo	02-10-2013	03-10-2013	5 horas	-Analista de Calidad	02-10-2013	-Analista de Calidad	
	Elaborar Acta de Pruebas	03-10-2013	04-10-2013	4 horas	-Analista de Calidad	04-10-2013	-Analista de Calidad	
	Codificar Módulo de Reportes de Asignación	04-10-2013	14-10-2013	3.5 días	-Programador Senior -Programador Junior	09-10-2013	-Programador Senior -Programador Junior	Adelanto de fechas
	Ejecutar Pruebas del Módulo	14-10-2013	16-10-2013	2 horas	-Analista de Calidad	13-10-2013	-Analista de Calidad	Adelanto de fechas
	Elaborar Acta de Pruebas	16-10-2013	16-10-2013	2 horas	-Analista de Calidad	14-10-2013	-Analista de Calidad	Adelanto de fechas
Módulo de Backups	Codificar Módulo de Backups	16-10-2013	23-10-2013	2.5 días	-Programador Senior -Programador Junior	16-10-2013	-Programador Senior -Programador Junior	Adelanto de fechas
	Ejecutar Pruebas del Módulo	24-10-2013	24-10-2013	3 horas	-Analista de Calidad	13-10-2013	-Analista de Calidad	Adelanto de fechas

	Elaborar Acta de Pruebas	24-10-2013	25-10-2013	2 horas	-Analista de Calidad	14-10-2013	-Analista de Calidad	Adelanto de fechas				
<b>MÉTRICAS DE CALIDAD EN EL PERÍODO</b>												
FACTOR RELEVANTE DE CALIDAD	OBJETIVO DE CALIDAD	MEDICIÓN DE MÉTRICA			OBSERVACIONES							
		FECHA	RESULTADO OBTENIDO									
Acta de Prueba	Obtener la aprobación de todos los elementos del módulo	30-09-2013 14-10-2013	Aprobado		No hubo ningún fallo al momento de ejecutar las pruebas de los módulos.							
<b>CONTROLES DE CALIDAD EN EL PERÍODO</b>												
ENTREGABLE	ESTÁNDAR DE CALIDAD APPLICABLE	OBSERVACIONES		OBSERVACIONES								
		FECHA	RESULTADO OBTENIDO									
Módulo de Comunidades, Reportes de Asignación y Backup's	Revisión de funciones y fechas de entrega	30-09-2013 14-10-2013	Aprobado	No hubo ningún fallo al momento de ejecutar las pruebas de los módulos.								
<b>ACTIVIDADES EN PROCESO A LA FECHA</b>												
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	FECHA DE FIN PROGRAMADA		FECHA DE FIN ESTIMADA	% DE AVANCE A LA FECHA		OBSERVACIONES					

Tabla 46 Estado de Avance del Proyecto N° 4

## **CAPÍTULO V: CIERRE DEL PROYECTO**

## 5.1. Gestión del proyecto

### 5.1.1. Cierre

#### A. Lecciones Aprendidas

Código de Proyecto	Proy-001		Nombre del Proyecto	Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal					
	Nro. De Referencia	Área / Categoría		Fecha	Amenaza / Oportunidad	Título	Descripción de la Situación	Descripción del Impacto en los objetivos del proyecto	Acciones Correctivas y Preventivas Implementadas
PROY-001-LEC-APRE-01	Cronograma	19/09/2013	Amenaza	Petición de cambio al cronograma	Se presentó la solicitud por parte del cliente de ejecutar cambios al cronograma inicial establecido, acortando la fecha de entrega del 6 de diciembre del 2013 original, al 21 de noviembre del 2013.	El adelantó de la fecha de entrega del proyecto supone un gran impacto en el cronograma, puesto que para cumplir con las fechas, habría que reducir los tiempos de las actividades, corriendo el riesgo de no poder realizar las	Se analizó y evaluó el efecto que podría tener en el desarrollo del proyecto. Puesto que al día siguiente se presentó una solicitud de eliminación de requerimientos, y por ende del desarrollo de un módulo, se pudo cumplir con	Verificar las fechas de término del proyecto a más detalle y consultar con el cliente si durante esas fechas no tiene actividades que puedan retrasar o detener el desarrollo del proyecto.	

					parroquia inicia diversas actividades por la aproximación de la navidad, y por ende se corría el riesgo de no poder ejecutar las pruebas finales y la aceptación para las fechas pactadas.	actividades a un 100%.	el adelanto de fechas que solicitó el cliente.	
PROY-001-LEC-APRE-02	Gestión de Requerimientos	20/09/2013	Oportunidad	Petición de cambio a los requerimientos iniciales del proyecto	<p>Se presentó la solicitud por parte del cliente de ejecutar cambios a los requerimientos que se habían pedido cuando se inició el proyecto.</p> <p>El cambio abarcaba la eliminación del módulo inicial de Eventos, el</p>	<p>La eliminación del requerimiento, y por lo tanto del módulo del sistema, dio oportunidad de poder incrementar los tiempos de codificación y de pruebas de los módulos, que se tenían inicialmente.</p>	<p>Se analizó y evaluó el efecto que podría tener en el desarrollo del proyecto.</p> <p>Puesto que la solicitud se hizo fechas antes de comenzar el desarrollo del dicho módulo, no tuvo gran impacto en el proyecto.</p>	Dedicar más tiempo y atención a los requerimientos iniciales del proyecto, indicando su nivel de prioridad e importancia.

					cual se debió al bajo nivel de prioridad que tuvo.	Considerando la petición anterior, esto ayudo a no tener que modificar en gran parte el cronograma, puesto que el tiempo que hubiera tomado en desarrollar y probar el módulo fue sustraído de la fecha de entrega del proyecto.	Los días de desarrollo y pruebas del módulo fueron sustraídos de los días de entrega del proyecto, por lo cual no hubo que cambiar en gran medida los tiempos de las actividades.	
PROY-001-LEC-APRE-03	Desarrollo del Software	19/07/2013	Amenaza	Fallo en las pruebas unitarias	Durante las pruebas que se hicieron con el líder de usuario al módulo de feligrés del sistema, este no cumplió con todas las funciones pues hubieron campos que no se tomaron en cuenta y por	Esta situación dio lugar a la corrección del módulo, lo cual demandó sacrificar un cierto periodo de tiempo de otra actividad.	Se tomaron en cuenta las observaciones y se solventaron los errores dando lugar a otra versión del acta de pruebas del módulo de feligrés.  No afectó en gran medida al desarrollo del siguiente	Realizar un checklist de todos los campos de cada módulo y comparar con los datos reales si estos cumplen con todos los requisitos.  Consultar con el cliente si hay situaciones especiales que puedan

					ende no se podían registrar los datos reales		módulo, el de "Sacramento s" puesto que había tiempo suficiente de holgura añadido a dicha actividad.	generar campos adicionales.
PROY-001-LEC-APRE-04	Diseño	06/07/2013	Amenaza	Inconformidad con el diseño propuesto	Durante el diseño de las interfaces se le hizo conocer al líder de usuario el modo en que las interfaces quedarían implementadas. Durante la muestra del diseño, el líder de usuario expresó su duda conforme al diseño, pidiendo interfaces más intuitivas.	Como la revisión del diseño se hizo dentro del plazo que se determinó para el mismo, el impacto no fue tan grande.  Se buscó libreras de diseño libres para dar más vida al diseño.	Se utilizaron 2 libreras libres de diseño, uno para el menú y otro para los botones, el resto del diseño consistió en agregar un color de fondo agradable a la vista del usuario.	Mostrar al usuario distintas propuestas de diseño final y anotar sus opiniones para acoplar el diseño del sistema a los gustos visuales del usuario.

PROY-001-LEC-APRE-05	Desarrollo del Software	11/09/2013	Amenaza	Falta de Control en las Actividades Programadas	No se tomó la debida atención al control de las fechas de entrega de los módulos para sus pruebas unitarias, perdiendo parte del tiempo del desarrollo en actualización documental	El impacto fue elevado puesto que se corría el riesgo de no cumplir con las fechas de entrega. De no haber cumplido con estas, se hubiera tenido que aumentar el tiempo asignado a algunas tareas o ampliado la fecha de entrega del producto.	La situación demandó esfuerzo extra para terminar los módulos dentro de las fechas establecidas. Aunque se logró cumplir con estas fechas, se tuvo que exigir más al equipo de desarrollo.	Llevar un recordatorio calendarizado de las fechas críticas durante la ejecución del proyecto, para balancear la carga documental con el desarrollo del sistema.
----------------------	-------------------------	------------	---------	-------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------

**B. Acta de Reunión de Cierre**

**Acta de Cierre de Proyecto**

<b>Nombre del proyecto:</b>	Implementación de un Sistema de Control de Sacramentos y Asignación de Personal en la parroquia San Clemente de Pisco	
<b>Área :</b>	<i>Secretaría</i>	
<b>Sede :</b>	<i>Parroquia San Clemente</i>	
<b>Fecha Emisión :</b>	<i>21 de Noviembre del 2013</i>	
<b>Participantes:</b>		
<b>Gerente de Proyecto :</b> Juan Guillermo Castillo Hernández	<b>Sponsor:</b> Párroco Vicente Diez Varona	<b>Líder de Usuarios:</b> Luisa Cajo Chacalizaza Secretaria
<p>Por medio de la presente se deja constancia de la aceptación por parte de la parroquia San Clemente del informe final del Proyecto de <i>Implementación de un Sistema de Control de Sacramentos y Asignación de Personal en la parroquia San Clemente de Pisco</i>.</p> <p>En este punto se da por concluido el proyecto, por lo que habiendo constatado la secretaría de la parroquia la finalización, entrega y aceptación del informe final del Proyecto de <i>Sistema de Control de Sacramentos y Asignación de Personal en la parroquia San Clemente de Pisco</i>.</p>		
 Vicente Diez <b>Sponsor</b>	 Juan Castillo <b>Gerente de Proyecto</b>	 Luisa Cajo <b>Líder de Usuarios</b>

### C. Certificado de Conformidad

## Certificado de Aceptación Funcional

<b>Nombre del proyecto:</b>	Implementación de un Sistema de Control de Sacramentos y Asignación de Personal en la parroquia San Clemente de Pisco
<b>Preparado por :</b>	<i>Juan Guillermo Castillo Hernández – Gerente del Proyecto</i>
<b>Fecha Emisión :</b>	<i>21 de Noviembre del 2013</i>

Por medio de la presente, la secretaría de la parroquia San Clemente de Pisco certifica que después de haber revisado todas las funcionalidades del **Sistema de Control de Sacramentos y Asignación de Personal**, desarrollado por el estudiante Juan Guillermo Castillo Hernández y patrocinado por el párroco Vicente Diez Varona, **cumple** con todas las funcionalidades solicitadas durante las actividades de levantamiento de información en concordancia con el documento oficial de requerimientos.

Es importante señalar que el sistema incluye todos los requerimientos solicitados por la secretaría de la parroquia y por el párroco luego de los cambios realizados a dichos requerimientos durante la ejecución del proyecto, logrando con esto una solución adaptada a los requerimientos de la organización en el mediano y largo plazo.

Emitimos este Certificado de Aceptación Funcional en señal de conformidad del servicio a solicitud del estudiante Juan Guillermo Castillo Hernández.

Pisco, 21 de Noviembre del 2013

  
\_\_\_\_\_  
Nombre: Vicente Diez Varona  
Cargo: Párroco

## 5.2. Ingeniería del proyecto

### 5.2.1. Pruebas

#### A. Plan de Pruebas

Nº	Módulo	Acciones	Técnicas	Dirigido a
1	Módulo Feligrés	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria
2	Módulo Sacramentos	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria
3	Módulo Reportes de Sacramentos	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria
4	Módulo Personal	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco
5	Módulo Comunidades	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco
6	Módulo Reportes de Asignaciones	Ingreso de datos reales al sistema. Prueba de distintos escenarios para determinar la eficiencia del módulo.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco
7	Módulo Backup's	Realización de copias de seguridad y posterior restauración	Pruebas backup's.	Secretaria

Tabla 47 Plan de Pruebas

## B. Informe de pruebas de Aceptación

Padre.

Vicente Diez Varona

Párroco de la parroquia San Clemente de Pisco

Con la presente se le informa a usted, que después de haber realizado las pruebas integrales del “Sistema de Control de Sacramentos y Asignación de Personal”, se le expresa lo siguiente:

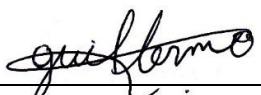
Que haciendo concluido las pruebas integrales a las interfaces del sistema, así como la relación entre ellas, se determinó que el sistema cumple con los requisitos expresados al iniciar el proyecto y, por lo tanto, queda como un resultado satisfactorio.

A continuación se mencionan las interfaces a las cuales se realizaron las pruebas:

- Interfaz de Países
- Interfaz de Ciudades
- Interfaz de Provincias
- Interfaz de Tipos (Personal y Comunidades)
- Interfaz de Lugares de Reunión
- Interfaz de Salvar Datos
- Interfaz de Restaurar Datos
- Interfaz de Feligrés
- Interfaz de Libro Sacramental
- Interfaz de Templos
- Interfaz de Bautismo
- Interfaz de Confirmación
- Interfaz de Matrimonio
- Interfaz de Actas
- Interfaz de Personal
- Interfaz de Comunidades
- Interfaz de Asignaciones
- Interfaz de Personal Asignado

Sin nada más que informar, me despido de usted, no sin antes agradecerle muy cordialmente su apoyo y disponibilidad para el desarrollo del proyecto.

Pisco, 18 de Noviembre del 2013



---

Juan G. Castillo Hernández  
Jefe de Proyecto

### 5.2.2. Implantación

#### A. Plan de capacitación

A continuación se describe, mediante un cuadro, el plan de capacitación que se llevó a cabo para que el usuario conozca el funcionamiento del sistema. Dicha capacitación se llevó a cabo los días viernes 15 al miércoles 20 del mes de Noviembre del año 2013.

Tema	Contenidos	Acciones	Técnicas	Dirigido a	Horas
1	Explicación detallada de las funciones del módulo Feligrés	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaría	1 h.
2	Explicación detallada de las funciones del módulo Sacramentos	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaría	3 h.
3	Explicación detallada de las funciones del módulo Reportes de Sacramentos	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaría	30 min.
4	Explicación detallada de las funciones del módulo Personal	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaría Párroco	1 h.
5	Explicación detallada de las funciones del módulo Comunidades	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolución de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaría Párroco	1 h.

		dudas.			
6	Explicación detallada de las funciones del módulo Reportes de Asignaciones	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolutión de dudas.	Asociación del manejo manual existente con el manejo del sistema. Pruebas de registros.	Secretaria Párroco	30 min.
7	Explicación detallada de las funciones del módulo Backup's	Explicación visual de las funciones de cada una de las interfaces del módulo. Prácticas dirigidas y absolutión de dudas.	Pruebas de backup's.	Secretaria	1 h.

Tabla 48 Plan de Capacitación

## B. Manual de Usuario

### Botones

A continuación se describen los botones que encontrará en las ventanas a lo largo del sistema.

#### Acciones Similares

Estos botones están presentes en la gran mayoría de ventanas. Todas realizan las mismas acciones.



NUEVO/CANCELAR => Botón que activa un nuevo registro y permite el ingreso de datos para guardarlos en el programa. También cancela el proceso de registro o modificación de datos.



MODIFICAR => Botón que activa el proceso de modificación de un registro, previamente registrado y seleccionado. Activa los datos que pueden ser modificados y deja inhabilitados aquellos que no pueden modificarse.



ELIMINAR => Botón que elimina un registro, previamente registrado y seleccionado. El registro queda eliminado de la base de datos del sistema.



GUARDAR => Botón que guarda la información de un nuevo registro o actualiza el de uno existente.



SALIR => Botón que cierra la ventana en la que se está trabajando.



BUSCAR => Botón que busca un determinado registro o inicializa una ventana de búsqueda.

## Acciones Propias



Cerrar Libro/Reaperturar => Botón que cierra (da de baja) un Libro Sacramental, previamente registrado y seleccionado o Reapertura un libro que ha sido cerrado manualmente. Solo se podrá reaperturar aquellos libros cuyos registros no sean iguales a la cantidad máxima de registros soportados.



Vista Previa => Botón que genera un archivo “pdf” de los registros a modo de reportes. Está presente en la ventana de “Actas de Sacramentos” y “Personal Asignado”. Lo que hace es generar un “pdf” de los datos visualizados para posteriormente poder imprimirlos físicamente.



Asignar => Botón que inicia el proceso de asignación del personal a una comunidad. Este proceso puede registrar o eliminar una asignación, según sea el caso.



Salvar Datos => Botón que genera una “copia de seguridad” de los registros del sistema y los guarda en un archivo, el cuál puede ser guardado en distintos dispositivos (USB, Internet, Discos Duros, etc...)



Restaurar Datos => Botón que “restaura” los registros de una copia de seguridad, previamente creada. Con esta función, los registros pueden ser restaurados a un estado anterior y recuperar todos los datos guardados en ese punto.

## Ventana Principal



## Cinta de Opciones del Menú

### Menú Básico

En este menú se encuentran las ventanas de función básicas, los cuáles dan soporte a las funciones del sistema y a las demás ventanas que controlan los registros.



Describiendo de izquierda a derecha:

1. Países => Ventana de mantenimiento de países.
2. Ciudades => Ventana de mantenimiento de ciudades.
3. Provincias => Ventana de mantenimiento de provincias (Perú).
4. Tipos => Ventana de mantenimiento de tipos (Personal y/o Comunidad).
5. Lugares => Ventana de mantenimiento de lugares de reunión (Comunidades).
6. Salvar => Ventana de creación de copias de seguridad (backup's) de los datos.
7. Restaurar => Ventana de restauración de copias de seguridad de los datos.

### Menú Control de Sacramentos

En este menú se encuentran las ventanas correspondientes al proceso de "Sacramentos" del sistema.



Describiendo de izquierda a derecha:

1. Feligrés => Ventana de gestión de Feligrés.
2. Libro => Ventana de mantenimiento de Libros Sacramentales.
3. Templos => Ventana de mantenimiento de Templos.
4. Bautismo => Ventana de gestión de Bautismos.
5. Confirmación => Ventana de gestión de Confirmaciones.
6. Matrimonio => Ventana de gestión de Matrimonios.
7. Actas => Ventana de generación de actas.

## Menú Asignación de Personal

En este menú se encuentran las ventanas correspondientes al proceso de “Asignación” del sistema.



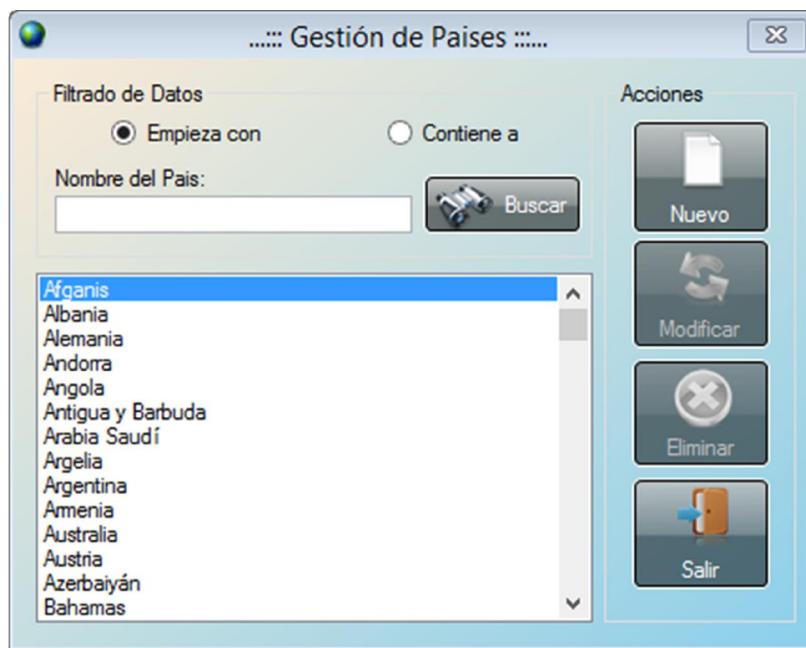
Describiendo de izquierda a derecha:

1. Personal => Ventana de gestión de personal.
2. Comunidad => Ventana de gestión de comunidades.
3. Asignación => Ventana de asignación del personal a comunidades.
4. Personal Asignado => Ventana de generación de reportes de asignación.

# Menú Básico

## Países

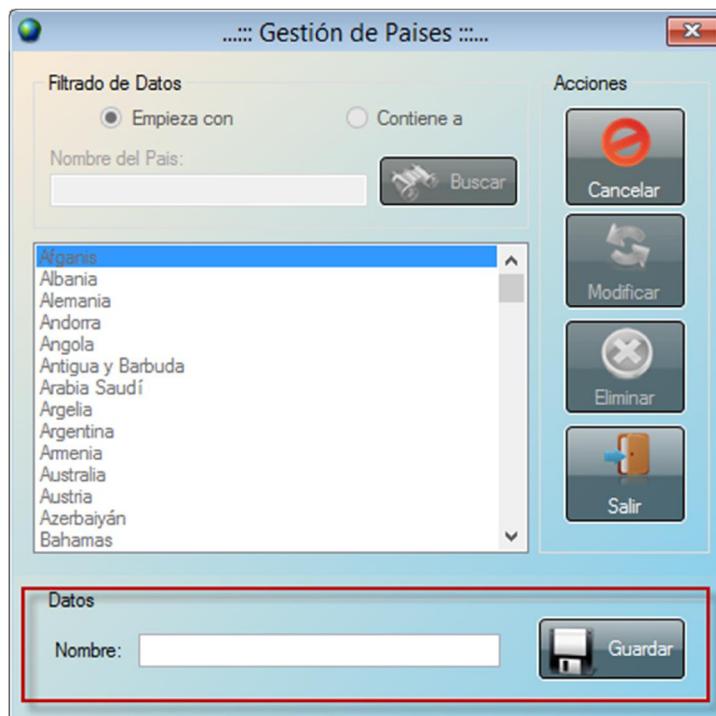
En esta ventana se pueden agregar nuevos países, modificar o eliminar países ya registrados. Por defecto, el sistema ya posee registros de países.



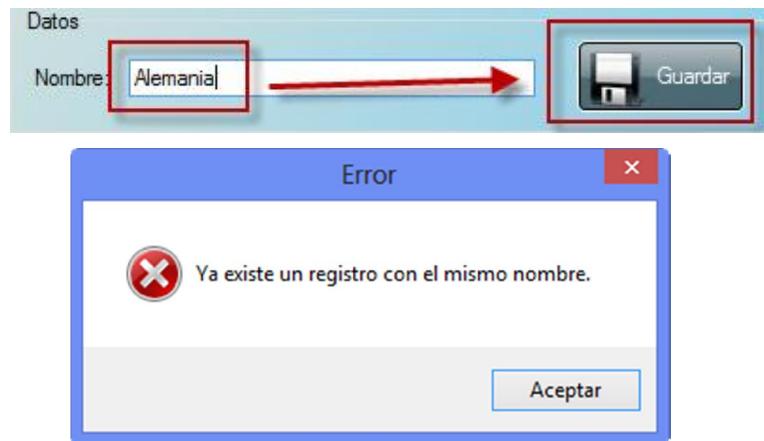
### Agregar un nuevo país

Para poder registrar un nuevo país, daremos click al botón “Nuevo”.

Al hacerlo, se desactivarán algunas opciones y se activarán las casillas para el registro del país.

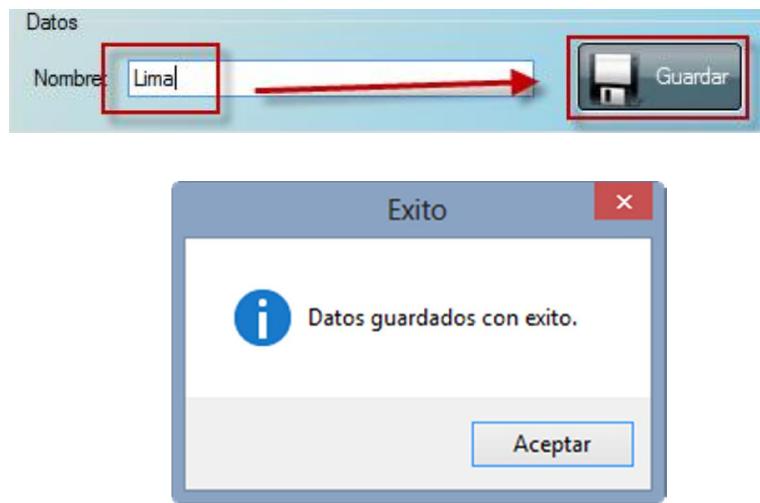


En la casilla que apareció debajo, colocamos el nombre del país que queremos ingresar y luego damos click en el botón “Guardar”.



En este caso, se quiso ingresar el país “Alemania” y al intentar guardarlo, nos muestra un mensaje indicándonos que ese país ya existe y por lo tanto no puede ser guardado.

Ahora intentaremos ingresar un nombre cualquiera.



### Buscar un país registrado

Se puede buscar haciendo click en el botón “Buscar” o presionando la tecla “Enter” en la casilla.

Hay 2 opciones para buscar a un país:

Empieza con => Buscará los países que empiecen con la letra o letras que se escriban en la casilla.

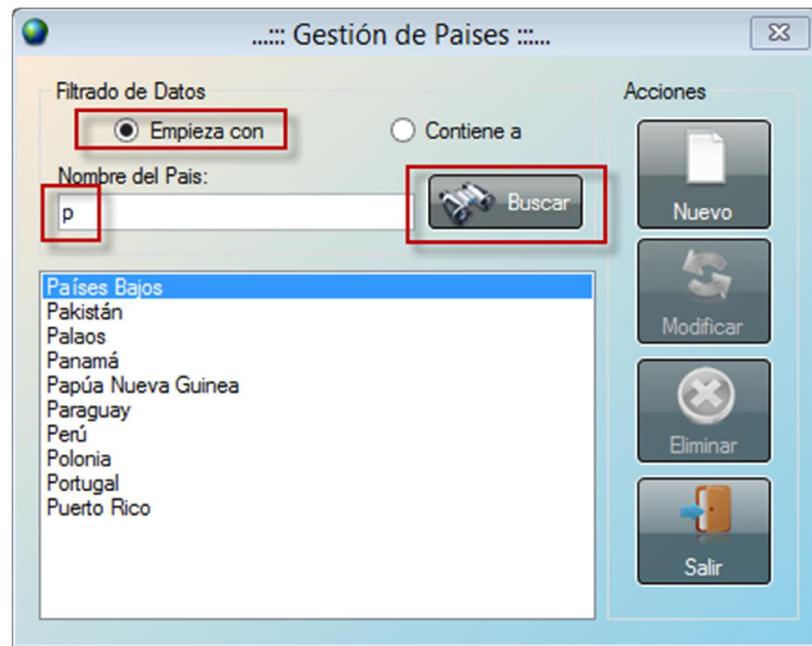
Contiene a => Buscará los países que contengan la letra o letras que se escriban en la casilla.

Filtrado de Datos

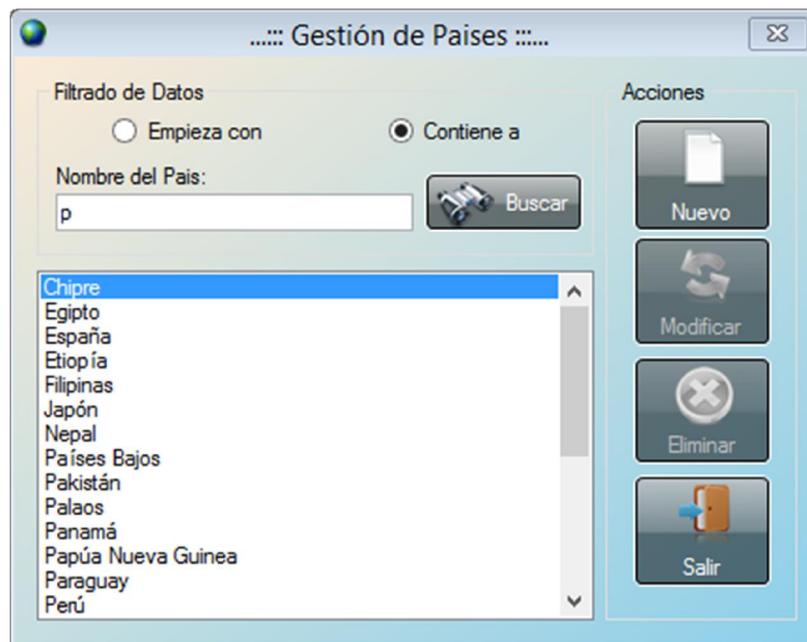
Empieza con       Contiene a

Nombre del País:

Por ejemplo si escribimos a letra “p” con la primera opción, buscará todos los países que empiecen por esa letra.



Si cambiamos a la segunda opción, buscará todos los países que contengan la letra “p” en sus nombres.

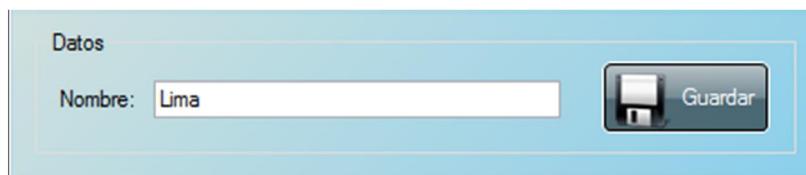


## Modificar el nombre de un país registrado

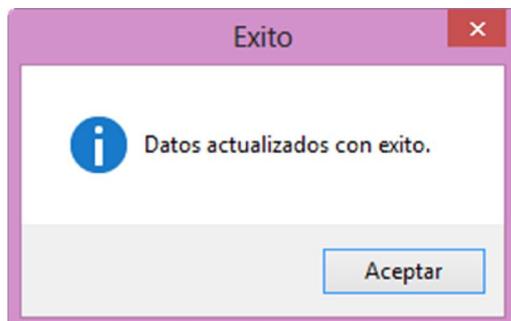
Para modificar el nombre de un país primero hay que buscarlo y seleccionarlo. Para este ejemplo buscaremos el registro que se ingresó al principio.



Observen que no es necesario colocar el nombre completo del país, solo las letras con las que empieza si seleccionamos la primera opción de búsqueda “Empieza con”. Hay que seleccionar un registro para activar las opciones “Modificar” y “Eliminar”. Luego damos click en “Modificar”.



Automáticamente se inhabilitaran ciertas opciones y podremos cambiar el nombre del país en la casilla que aparece debajo. Una vez cambiado el nombre daremos click en “Guardar” para cambiar el nombre.

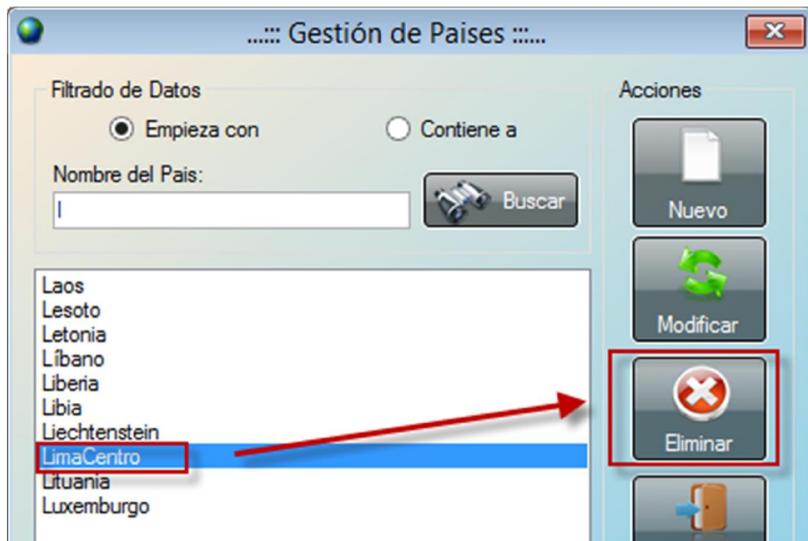


NOTA: Si no seleccionamos un registro haciendo click sobre él, las opciones “Modificar” y “Eliminar” no se habilitarán.

## Eliminar un país

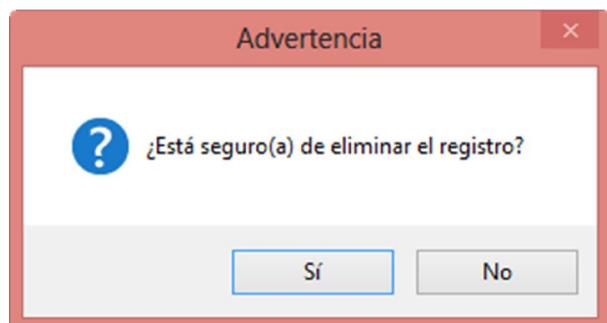
Para eliminar un país, se debe buscar y seleccionar el registro, al igual que se hizo para modificar.

OJO: Sólo se pueden eliminar registros nuevos. No podrá eliminar los registros que están por defecto en el sistema.



Una vez que hemos seleccionado el registro que queremos eliminar, hacemos click en el botón “Eliminar”.

Al hacerlo, se mostrará un mensaje de advertencia, el cual nos pedirá confirmar que realmente se desea eliminar el registro. Esto es en caso de que se halla hecho click en el botón por accidente.



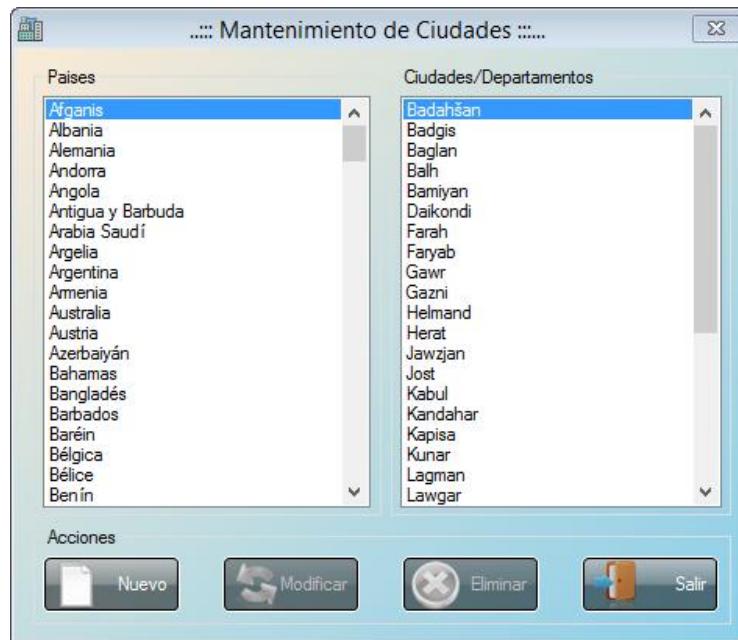
## Ciudades

En esta ventana se pueden agregar nuevas ciudades, modificar o eliminar ciudades ya registradas. Estas ciudades pertenecen a los países, es por ello que se muestran 2 listas:

- Los países, al lado izquierdo.
- Las ciudades, al lado derecho.

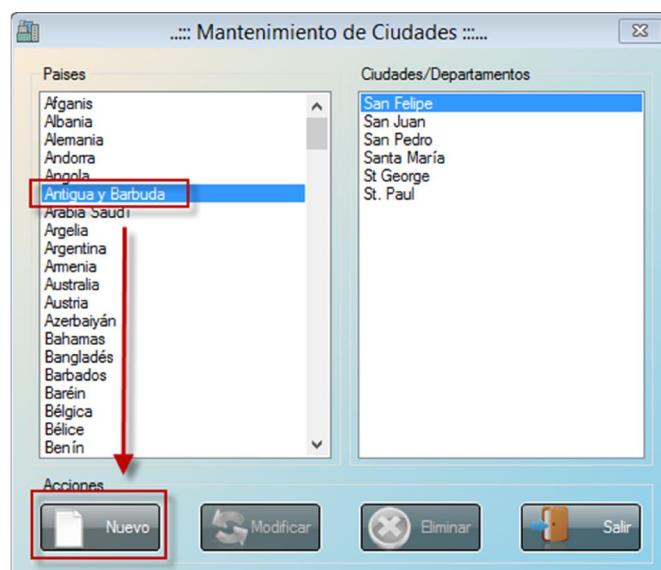
Al hacer click en un país, automáticamente la lista de la derecha mostrará sus ciudades.

Por defecto, el sistema ya posee registros de países

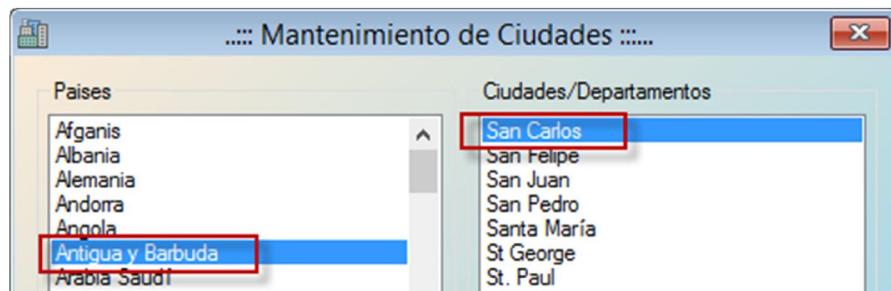
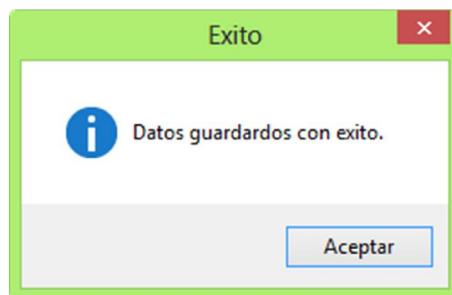
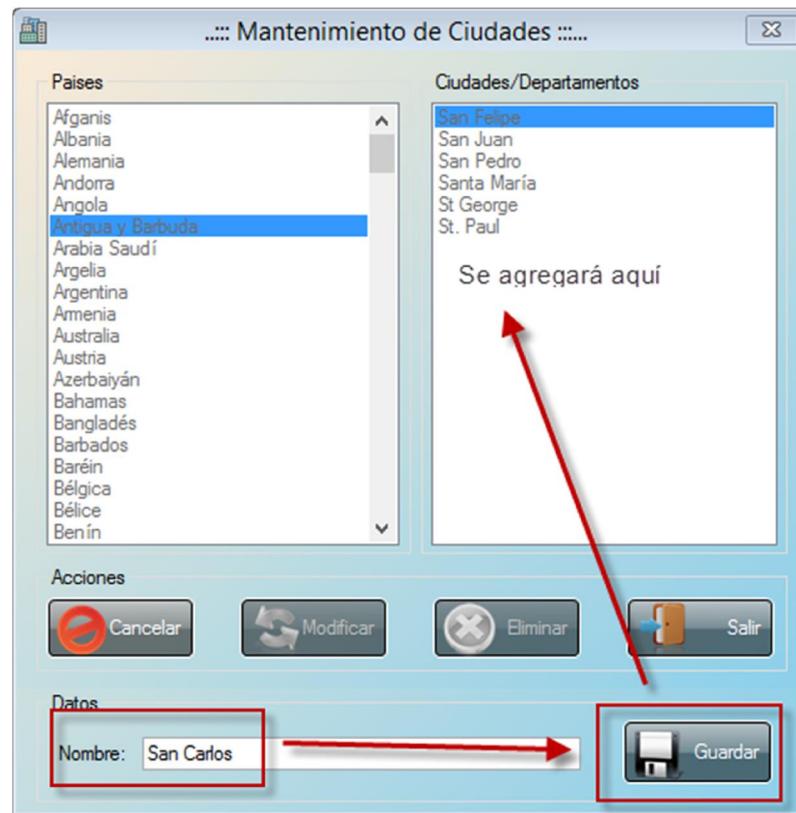


### Agregar una nueva ciudad

Para agregar una nueva ciudad, primero seleccionaremos el país al cuál queremos agregarle la ciudad. Luego daremos click en el botón “Nuevo”.

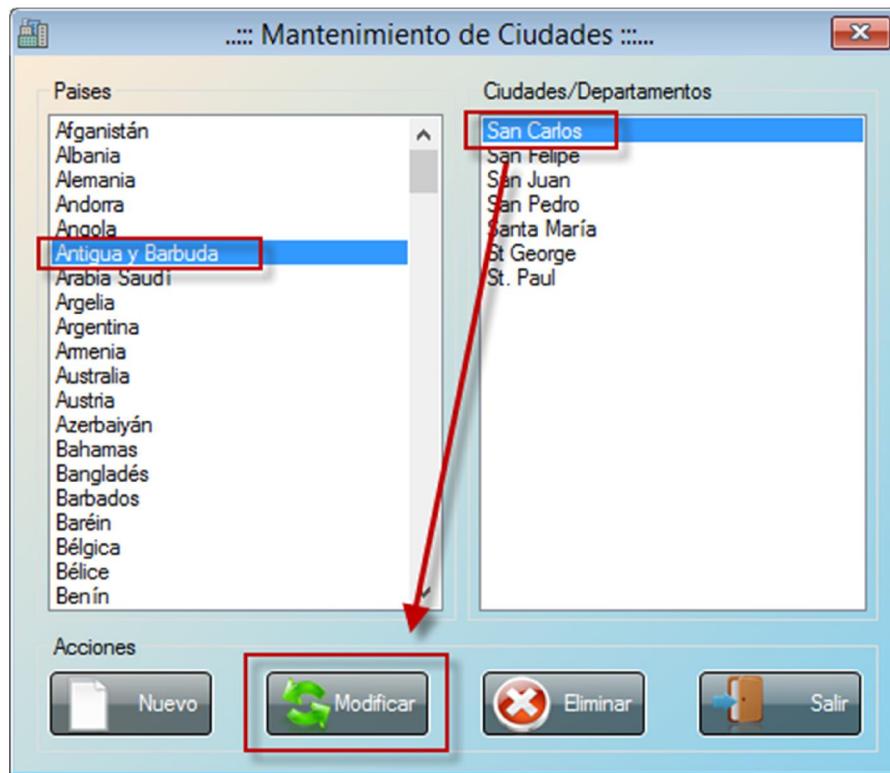


Una vez hecho, escribiremos el nombre de la ciudad que queremos registrar y hacemos click en "Guardar".



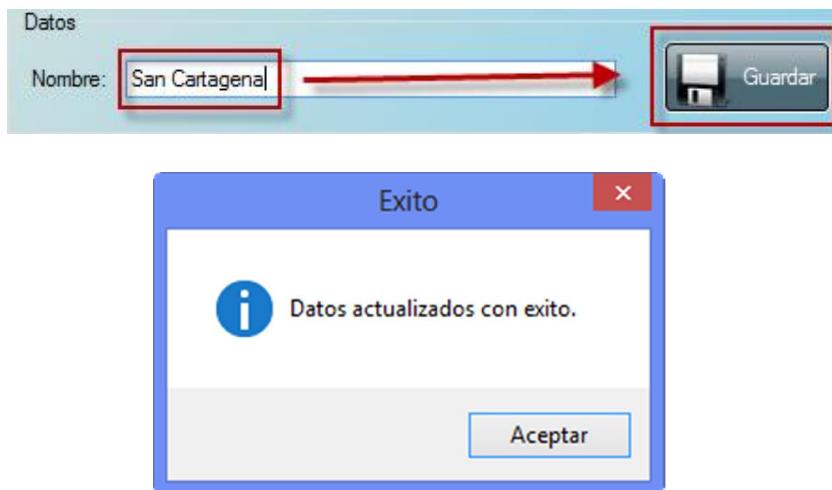
## Modificar el nombre de una ciudad registrada

Para modificar el nombre de una ciudad, primero debemos seleccionarla para luego hacer click en el botón “Modificar”.



Al hacerlo, se desactivaran algunas opciones y se mostrará el nombre de la ciudad en la parte inferior.

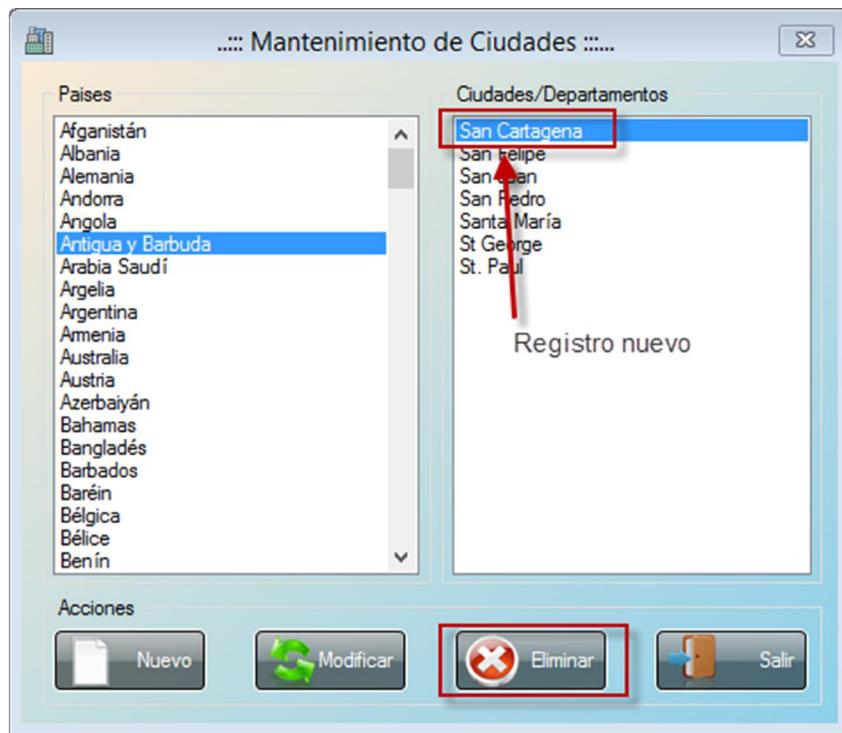
Simplemente modificamos el nombre y luego damos click en “Guardar”.



## Eliminar una ciudad

Para eliminar una ciudad, debemos seleccionarla y luego dar click en el botón “Eliminar”.

NOTA: No se podrá eliminar los registros por defecto del sistema. Solo se pueden eliminar las ciudades que el usuario ingrese.



Al hacer click, se mostrará un mensaje de advertencia para confirmar si realmente se desea eliminar el registro seleccionado.

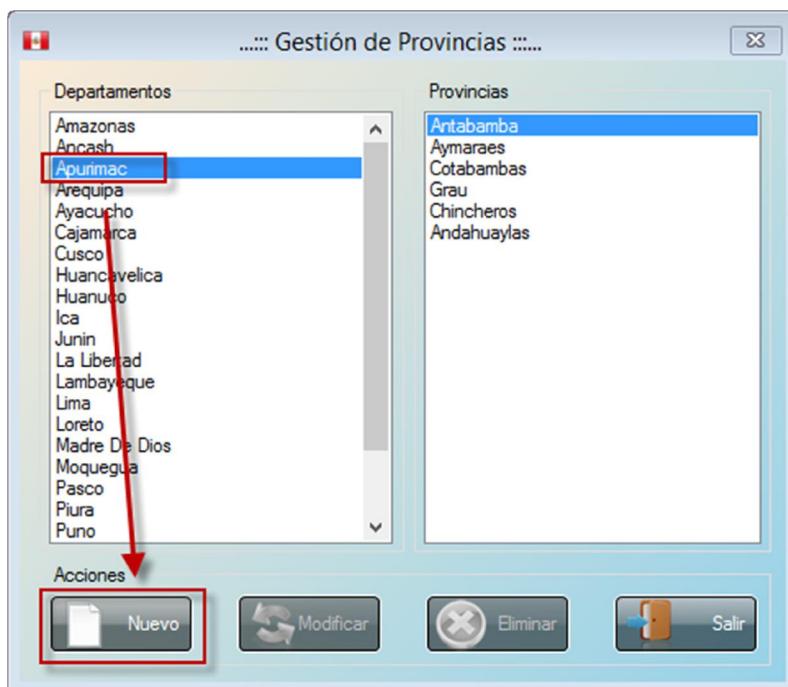
## Provincias

Esta ventana sirve para agregar, modificar o eliminar provincias del país “Perú”. La forma de agregar, modificar y eliminar son exactamente iguales que en la ventana “Ciudades”.

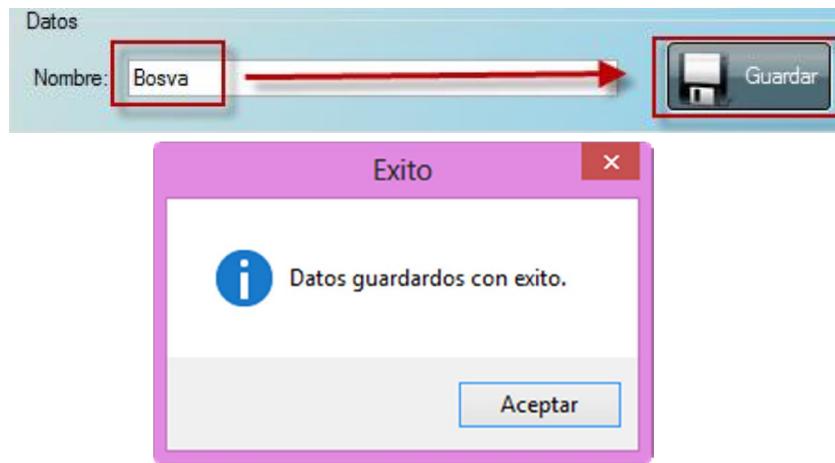


### Agregar una nueva provincia

Para agregar una provincia, seleccionamos un departamento del lado derecho y luego damos click en “Nuevo”.

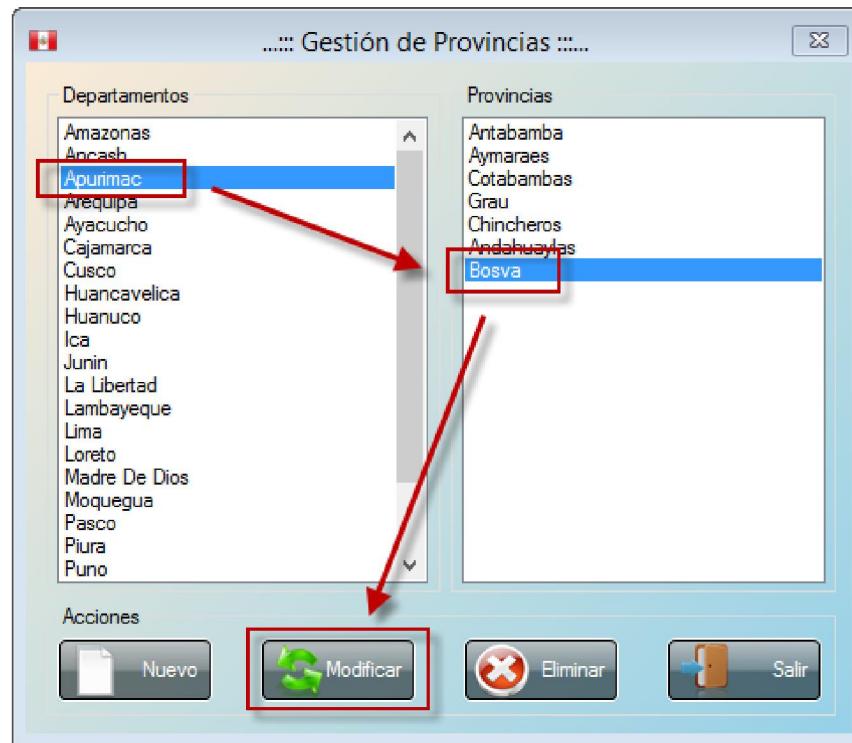


Luego ingresamos el nombre en la casilla inferior que aparece y damos click en “Guardar”.



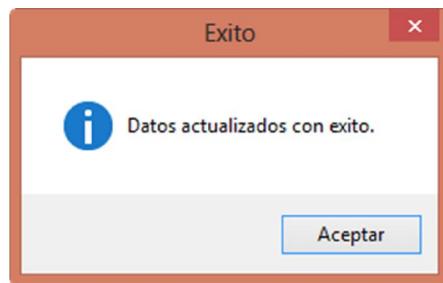
### Modificar el nombre de una provincia registrada

Para modificar el nombre de una provincia primero debemos seleccionarlo y luego hacer click en “Modificar”.



Ahora solo procedemos a modificar el nombre de la provincia y damos click en “Guardar”.

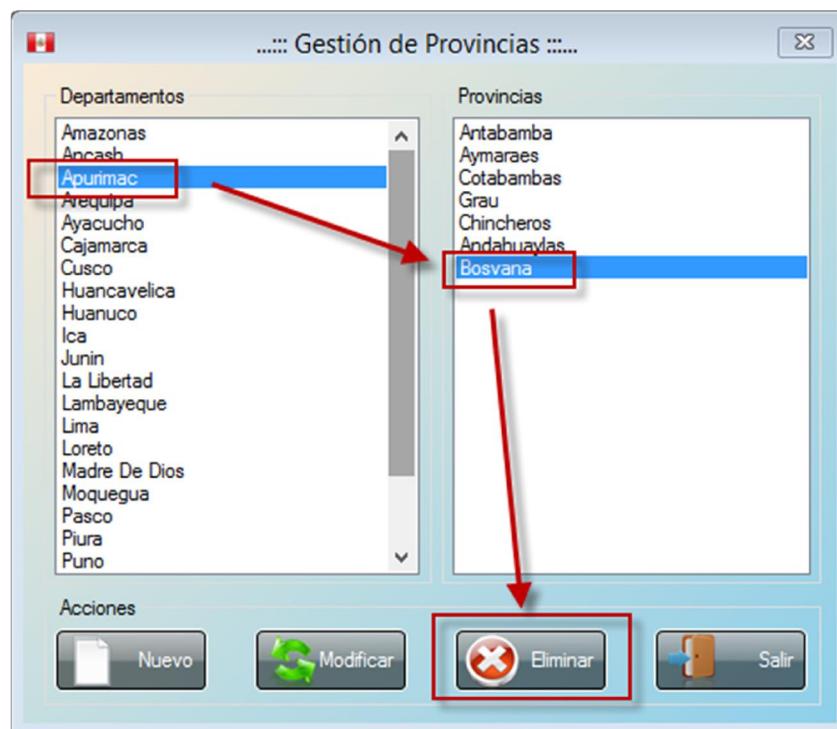




## Eliminar una provincia

Para eliminar una provincia, primero debemos seleccionarla y luego dar click en el botón "Eliminar".

NOTA: Sólo se puede eliminar las provincias que sean registradas por el usuario, no podrá eliminar las que vienen por defecto en el sistema.

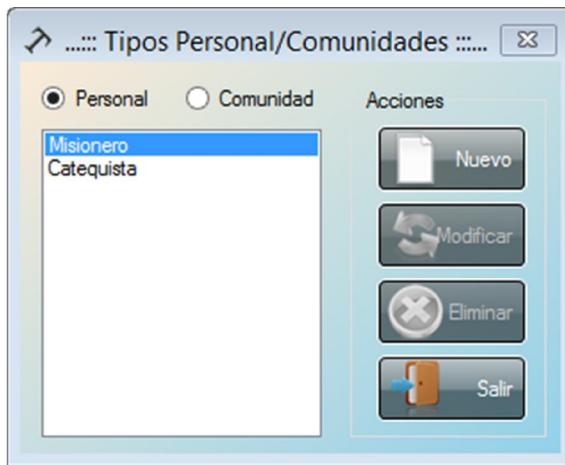


## Tipos

Esta ventana permite agregar o modificar los tipos de personal y comunidad que hay en el sistema. Por defecto vienen 2 tipos para cada uno:

- Comunidad
  - Grupo Parroquial
  - Comunidad Eclesial de Base
- Personal
  - Misionero
  - Catequista

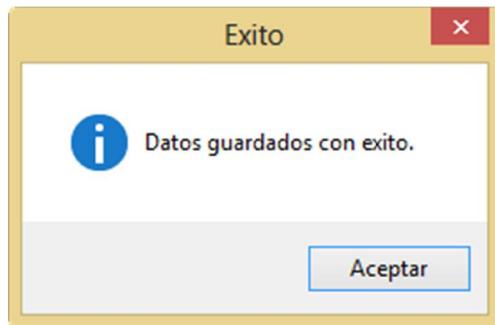
Estos registros no podrán ser eliminados, pero si modificados.



## Agregar un nuevo tipo

Para agregar un nuevo tipo, solo debemos dar click en botón “Nuevo”. Luego colocamos el nombre del tipo y seleccionamos si va a ser de personal o de comunidad. Por último damos click en “Guardar”.





NOTA: No se podrá cambiar si el tipo es de personal o de comunidad.

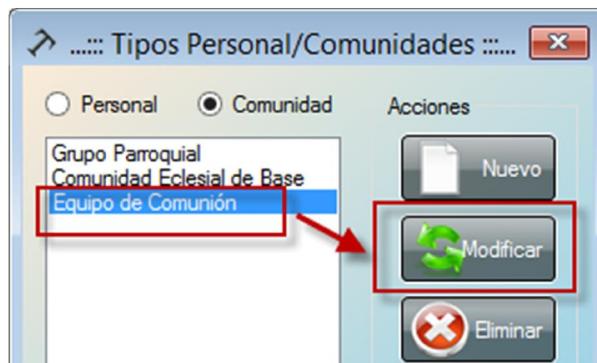
## Buscar un tipo

La búsqueda de los tipos solo está disponible seleccionando si es de personal o de comunidad.

Two side-by-side search interface windows. Both have a title bar "...:: Tipos Personal/Comunitad::...".  
Left window: Radio buttons for "Personal" (selected) and "Comunidad". Below is a list box containing "Misionero" and "Catequista".  
Right window: Radio buttons for "Personal" and "Comunidad" (selected). Below is a list box containing "Grupo Parroquial", "Comunidad Eclesial de Base", and "Equipo de Comunión".

## Modificar un tipo

Para modificar un tipo primero debemos seleccionarlo y luego dar click en el botón "Modificar".



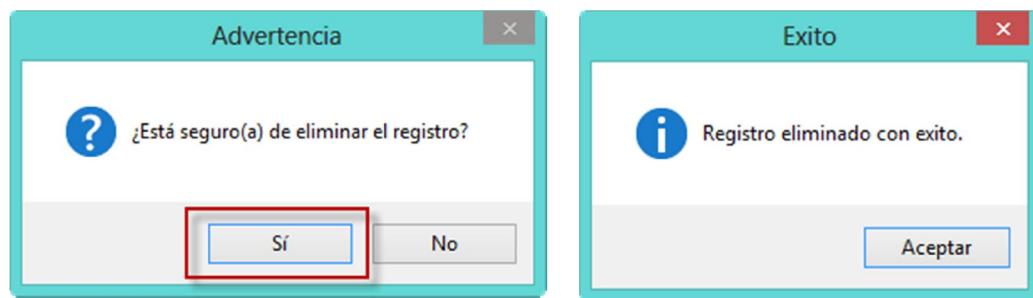
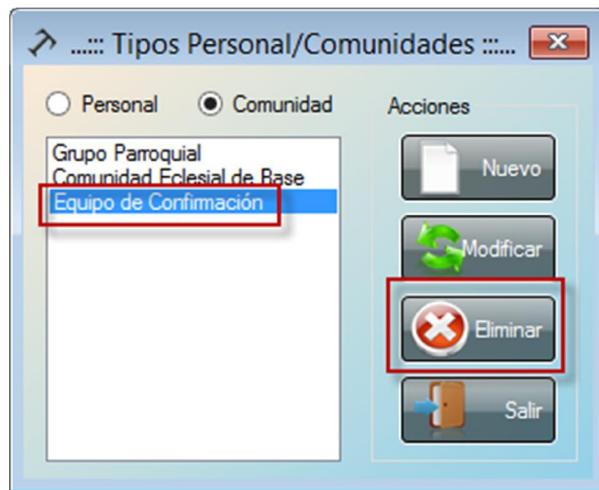
Luego modificamos el nombre y/o si es de personal o comunidad y damos click en "Guardar".



## Eliminar un tipo

Para eliminar un tipo, solo debemos seleccionarlo y dar click en el botón "Eliminar".

NOTA: No se podrá eliminar los registros que vienen por defecto en el sistema. Sólo se podrán eliminar los nuevos registros hechos por el usuario.



## Lugares

En esta ventana podremos agregar, modificar o eliminar los lugares de reunión de las comunidades.



## Agregar un lugar

Para agregar un lugar debemos dar click en el botón “Nuevo”.

Luego debemos ingresar el nombre del lugar y su dirección (opcional) y dar click en “Guardar”.

The top part of the image shows a 'Datos' form with two fields: 'Nombre:' containing 'Templo Principal' and 'Direccion:' containing 'S/N'. A red box highlights the 'Nombre:' field, and a red arrow points from it to the 'Guardar' button, which is also highlighted with a red box. The bottom part shows a confirmation dialog box titled 'Datos' with a blue information icon and the message 'Datos guardados con éxito.' (Data saved successfully.). It has an 'Aceptar' (Accept) button at the bottom.

## Buscar un lugar

Para buscar un lugar, sólo debemos colocar las primeras letras de su nombre y dar click en el botón “Buscar” o presionar la tecla “Enter”.

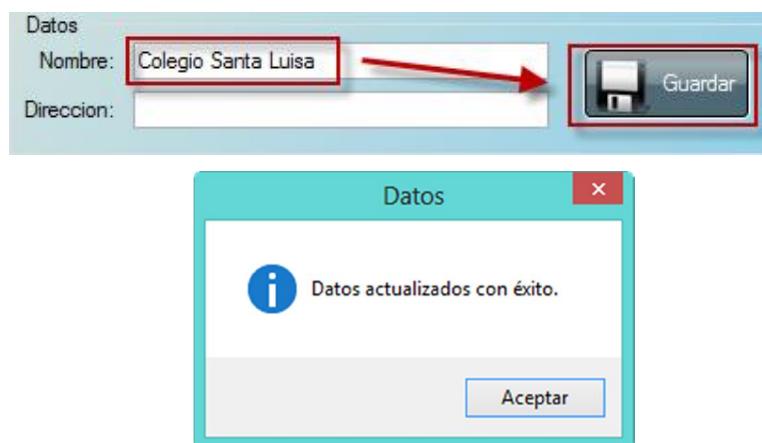


## Modificar un lugar

Para modificar un lugar, primero tenemos que buscarlo, seleccionarlo y luego dar click en el botón “Modificar”.



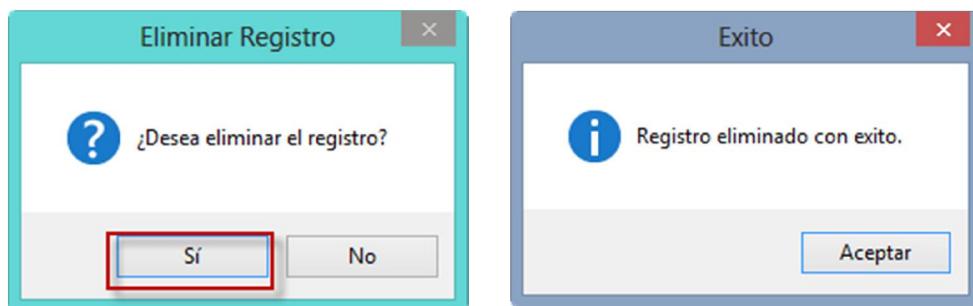
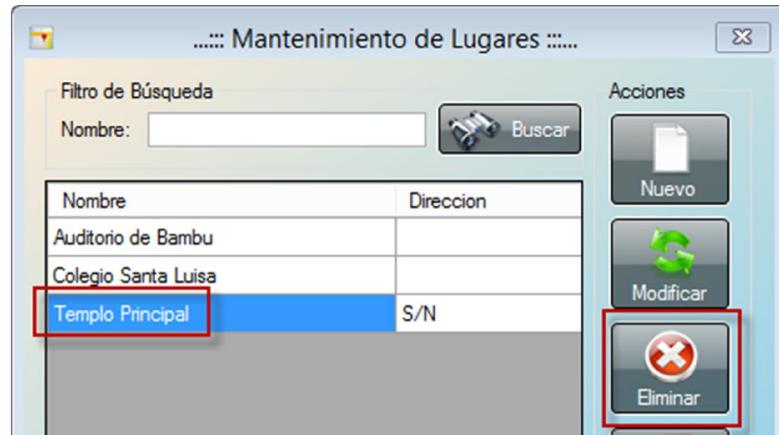
Luego podemos modificar su nombre y/o dirección en la parte inferior. Para guardar los cambios damos click en “Guardar”.



## Eliminar un lugar

Para eliminar un lugar, solo debemos seleccionarlo y hacer click en el botón “Eliminar”.

NOTA: No se podrá eliminar los registros que vienen por defecto en el sistema. Sólo se puede eliminar los nuevos registros que ingrese el usuario.

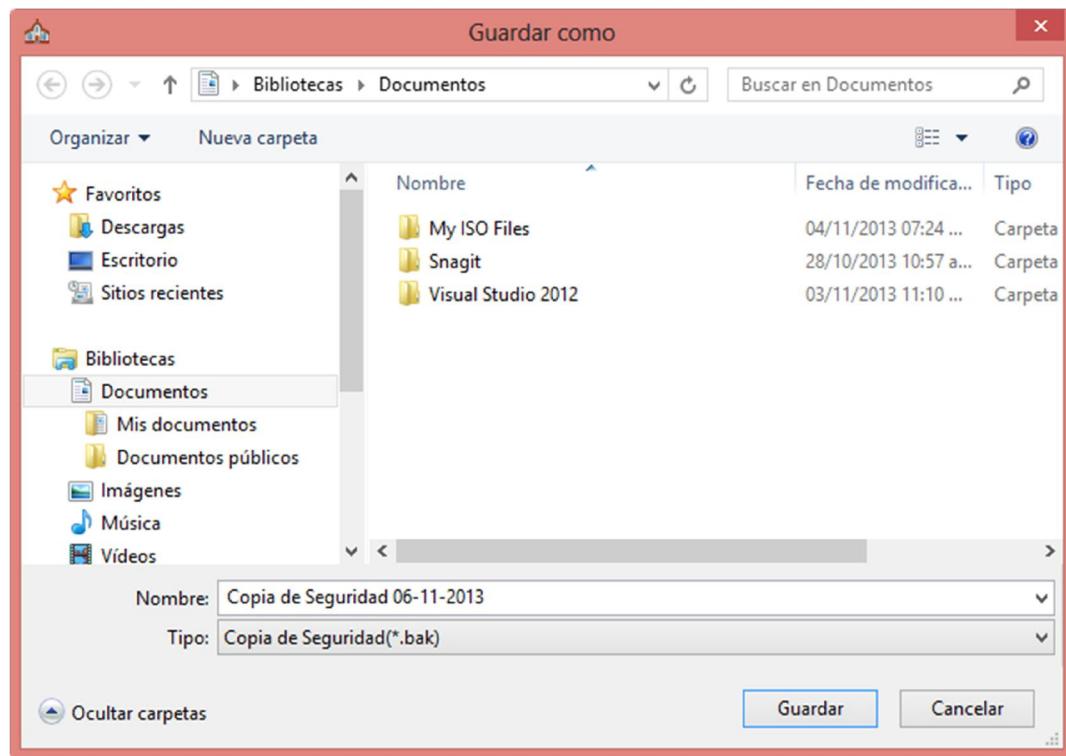


## Salvar

Esta ventana crea un archivo de “backup” de todos los registros del sistema, es decir, crea un archivo con todo lo que se ha ingresado en el sistema. Dicho archivo puede transportarse por unidades USB, guardarlo en otro disco duro, o llevarlo a la nube para su mayor protección. Es te archivo es llamado de “respaldo” y servirá para poder proteger los datos en caso de alguna falla del sistema operativo.



Para iniciar el proceso de salvado de los datos sólo debemos hacer click en el botón “Salvar Datos”.



Al hacerlo aparecerá un cuadro donde elegiremos la ubicación del archivo que será generado, su nombre y el tipo de archivo.

Una vez escogido la ruta, el tipo de archivo y el nombre del mismo hacemos click en “Guardar”. Automáticamente empezará el proceso de salvado de datos. Esto puede tomar algunos minutos.

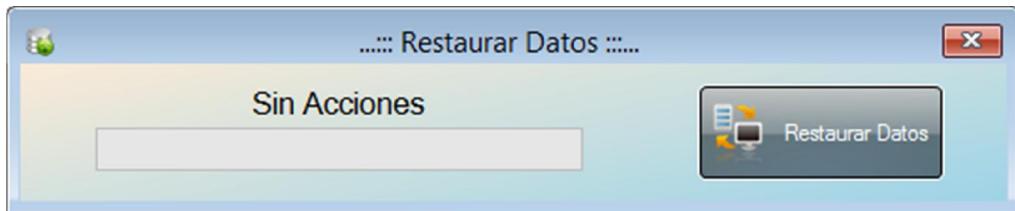


Exploramos en la ubicación que seleccionamos y comprobamos que el archivo fue generado con éxito.

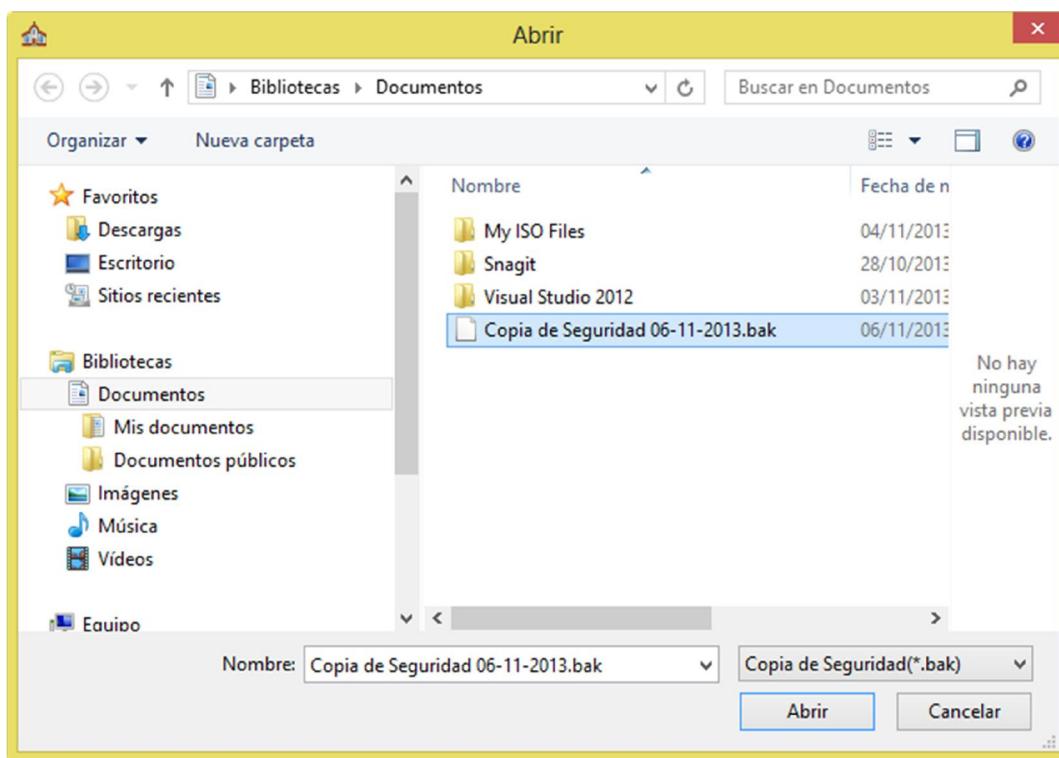
Nombre	Tipo	Tamaño
My ISO Files	Carpeta de archivos	
Snagit	Carpeta de archivos	
Visual Studio 2012	Carpeta de archivos	
<b>Copia de Seguridad 06-11-2013.bak</b>	Archivo BAK	108 KB

## Restaurar

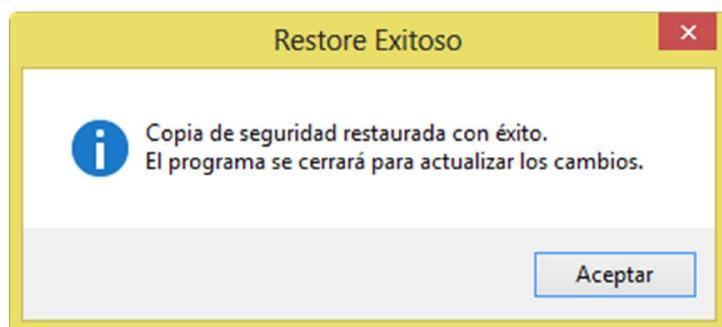
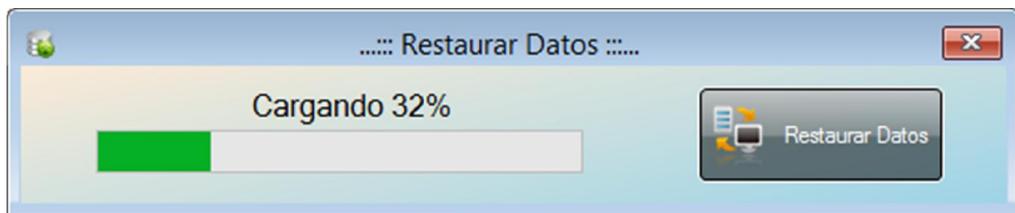
Esta ventana sirve para restaurar los archivos del sistema a partir de una copia de seguridad generada con anterioridad. Por ejemplo, si el sistema operativo sufre una infección de virus fuerte y se tuvo que formatear todo el sistema, una vez instalado el sistema todos los archivos que se hayan guardado se habrán perdido. Con esta ventana se puede recuperar todos los archivos guardados y que estén protegidos en un archivo backup's (generado en la ventana anterior).



Para iniciar el proceso solo tenemos que dar click en el botón “Restaurar Datos”.



En la ventana que aparece seleccionamos el archivo de backup que generamos en la ventana “Salvar” y damos click en “Abrir”.



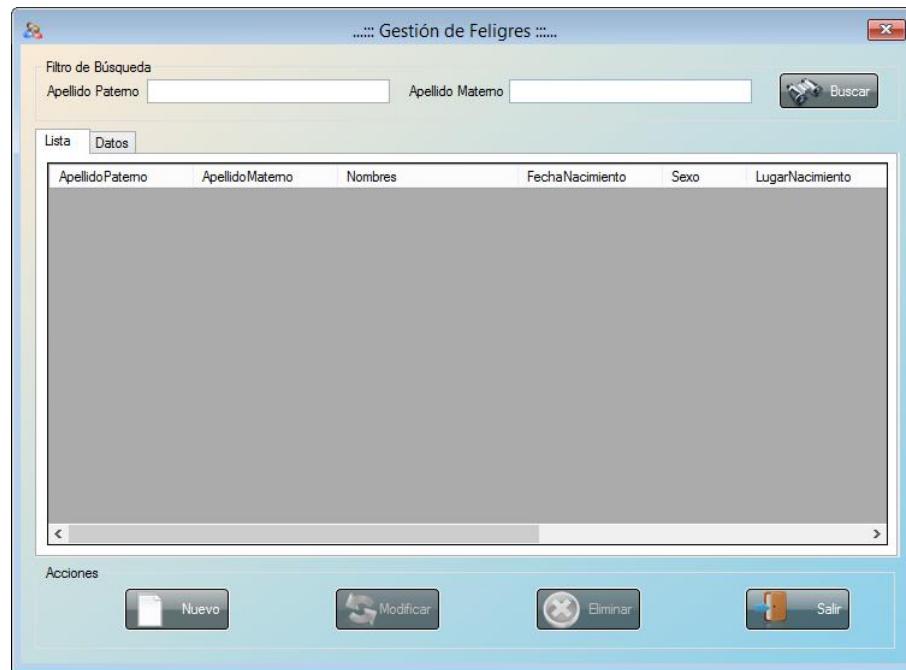
Una vez completado el proceso se mostrará un mensaje indicándonos que el proceso fue exitoso y que el sistema se cerrará para actualizar los cambios.

Una vez que abramos nuevamente el sistema, los archivos se habrán restaurado.

# **Menú Control de Sacramentos**

## Feligrés

En esta ventana podremos agregar, modificar o eliminar a un feligrés.



### Agregar un feligrés

Para agregar un nuevo feligrés hacemos click en el botón “Nuevo”. Al hacerlo, cambiará la pestaña para poder ingresar los datos del feligrés.

A screenshot of the same application window, but the "Datos" tab is selected. This view contains several data entry fields grouped into sections: "Datos Personales" (Apellido Paterno, Fecha de Nacimiento, Apellido Materno, Sexo, Nombres), "Lugar de Nacimiento" (Departamento, Provincia, Distrito, Nacional checkbox), "Datos del Padre" (Padre: Nombres y Apellidos, Departamento, Provincia, Distrito, Nacional checkbox), and "Datos de la Madre" (Madre: Nombres y Apellidos, Departamento, Provincia, Distrito, Nacional checkbox). At the bottom left is a checkbox "Existe como Personal" and a "Guardar" button. The bottom toolbar includes "Cancelar" (Cancel), "Modificar" (Modify), "Eliminar" (Delete), and "Salir" (Exit).

En total hay 4 divisiones en los datos:

- Datos Personales del Feligrés
- Lugar de Nacimiento del Feligrés
- Datos del Padre
- Datos de la Madre

Empezaremos llenando los datos en ese mismo orden. Primero llenamos los datos personales del feligrés.

Datos Personales

Apellido Paterno:	Fecha de Nacimiento:
Jacobo	24/09/1990 <input type="button" value=""/>
Apellido Materno:	Sexo:
Canelo	Masculino <input type="button" value=""/>
Nombres:	
Alembert Gustavo	

Debemos tener en cuenta que la fecha de nacimiento está restringida a la fecha actual como máximo. Suponiendo que la fecha en la que estamos llenando los datos sea "12/10/2013" y queremos ingresar la fecha de nacimiento "21/08/1992", debemos ingresar primero el año para que reconozco que estamos ingresando una fecha anterior ya que si colocamos primero el día, se devolverá al valor de la fecha actual por estar fuera de rango.

Ahora seguiremos ingresando el lugar de nacimiento del feligrés. Para esto hay 2 opciones designados por la casilla de verificación "Nacional". Al estar marcado, estamos asignando que nació en "Perú" y por lo tanto se muestran los departamentos con las provincias, y el distrito que es opcional. Por el contrario, si el feligrés nació en un país extranjero, se desmarca la casilla y automáticamente se mostrarán los países con sus ciudades.

Lugar de Nacimiento

Departamento:	Amazonas <input type="button" value=""/>
Provincia:	Chachapoyas <input type="button" value=""/>
Distrito:	<input type="text"/> <input checked="" type="checkbox"/> Nacional

Lugar de Nacimiento

País:	Afganistán <input type="button" value=""/>
Ciudad:	Badahšán <input type="button" value=""/>
<input type="checkbox"/> Nacional	

Ahora seguiremos ingresando los datos del padre.

Para esto hay 2 opciones a parte del lugar de nacimiento del padre, asignado por la casilla de verificación "No hay padre".

Si el feligrés no tiene padre o no es reconocido por este, marcamos la casilla de verificación. Al hacerlo, toda la zona de los datos del padre se inhabilitará automáticamente.

Datos del Padre

Padre: (Nombres y Apellidos)	<input checked="" type="checkbox"/> No hay padre
Departamento:	Amazonas
Provincia:	Chachapoyas
Distrito:	<input checked="" type="checkbox"/> Nacional

Por el contrario, si el feligrés si posee padre, se desmarca la casilla y se procede a ingresar sus datos.

Datos del Padre

Padre: (Nombres y Apellidos)	<input type="checkbox"/> No hay padre
Roberto Alonso Jacobo Solier	
Departamento:	Ica
Provincia:	Pisco
Distrito:	<input checked="" type="checkbox"/> Nacional

De la misma forma se llenan los datos de la madre.

NOTA: Obligatoriomente esta zona debe llenarse.

Datos de la Madre

Madre: (Nombres y Apellidos)	Rosa Angelica Canelo Canelo
Departamento:	Ica
Provincia:	Ica
Distrito:	<input checked="" type="checkbox"/> Nacional

Como ya tenemos todos los datos listos, hacemos click en el botón “Guardar”.

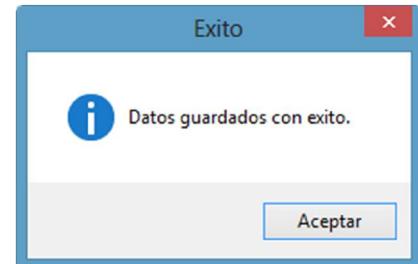
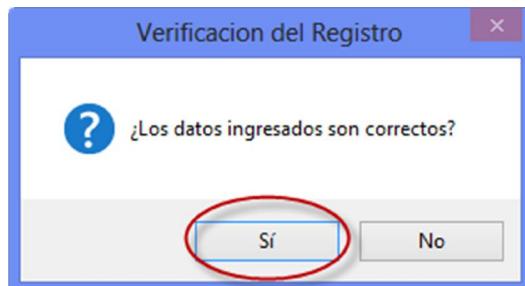


NOTA: Si no llenamos los datos aparecerá un mensaje de error.

El campo distrito es opcional, por lo tanto no es necesario llenarlo.

<b>Datos Personales</b> Apellido Paterno: <input type="text" value="Jacobo"/> Apellido Materno: <input type="text" value="Canelo"/> Nombres: <input type="text" value="Alemberg Gustavo"/>	<b>Datos del Padre</b> Padre: (Nombres y Apellidos) <input type="text" value="Roberto Alonso Jacobo Solier"/> <input type="checkbox" value="No hay padre"/> Departamento: <input type="text" value="Ica"/> Provincia: <input type="text" value="Pisco"/> Distrito: <input type="text"/> <input checked="" type="checkbox" value="Nacional"/>
<b>Lugar de Nacimiento</b> Departamento: <input type="text" value="Ica"/> Provincia: <input type="text" value="Ica"/> Distrito: <input type="text"/> <input checked="" type="checkbox" value="Nacional"/>	
<input type="checkbox" value="Existe como Personal"/> <input style="background-color: #ccc; border: 1px solid black; padding: 2px 10px; border-radius: 5px; width: 100px; height: 30px;" type="button" value="Guardar"/>	

Primero se muestra una ventana de confirmación que nos pregunta si los datos que hemos ingresados son correctos.



Nombres	FechaNacimiento	Sexo	LugarNacimiento	Padre
ALEMBERT GUSTAVO	24/09/1990	Masculino	Ica, Ica	Roberto Alonso Jacobo Solier

Una vez guardado el registro, automáticamente se cambiará a la pestaña lista donde están los registros.

NOTA: Como se puede observar, Los apellidos y nombres del feligrés están en mayúsculas, esto es para tener una uniformidad al momento de guardar los datos, es decir, aun si colocamos los apellidos y nombres todo en minúsculas, el sistema lo guardará todo en mayúsculas.

De la misma forma ocurre con el padre y la madre, el sistema guardara los nombres y apellidos con la primera letra de cada palabra en mayúsculas, aún si se escribe todo en minúsculas.

Otra forma de registrar a un feligrés, es si este ya está registrado como personal. En este caso utilizaremos la casilla “Existe como Personal”, al marcarla se deshabilitaran las casillas de la zona “Datos Personales” y podremos buscar al personal con el botón “Buscar” que aparece.

Existe como Personal

Datos Personales

Apellido Paterno:	Fecha de Nacimiento:
<input type="text"/>	06/11/2013 <input type="button" value=""/>
Apellido Materno:	Sexo:
<input type="text"/>	Masculino <input type="button" value=""/>
Nombres:	<input style="outline: none; border: 2px solid red; border-radius: 5px; padding: 2px 10px;" type="button" value="Buscar"/>

Al hacerlo se abrirá una nueva ventana donde buscaremos al personal, lo seleccionamos y luego presionamos la tecla “Enter”.

..... Lista del Personal .....

Filtro de Búsqueda

Apellido Paterno	Apellido Materno	Apellidos	Fecha Nacimiento	Sexo	Tipo	Domicilio
4	Soto	Borges	Paul Edson	15/07/1985	Masculino	1

Datos Personales

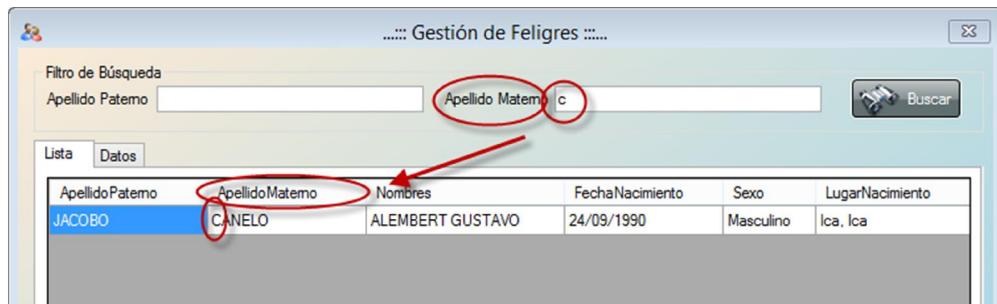
Apellido Paterno:	Fecha de Nacimiento:
Soto	15/07/1985 <input type="button" value=""/>
Apellido Materno:	Sexo:
Borges	Masculino <input type="button" value=""/>
Nombres:	<input style="outline: none; border: 2px solid red; border-radius: 5px; padding: 2px 10px;" type="button" value="Buscar"/>

Como se puede observar, todos los datos de esta zona se han llenado de manera automática, de esta manera el sistema reconoce que esa personal está registrada como feligrés y como personal, haciendo que se pueda registrar sin tener que llenar todos sus datos nuevamente.

## Buscar un feligrés

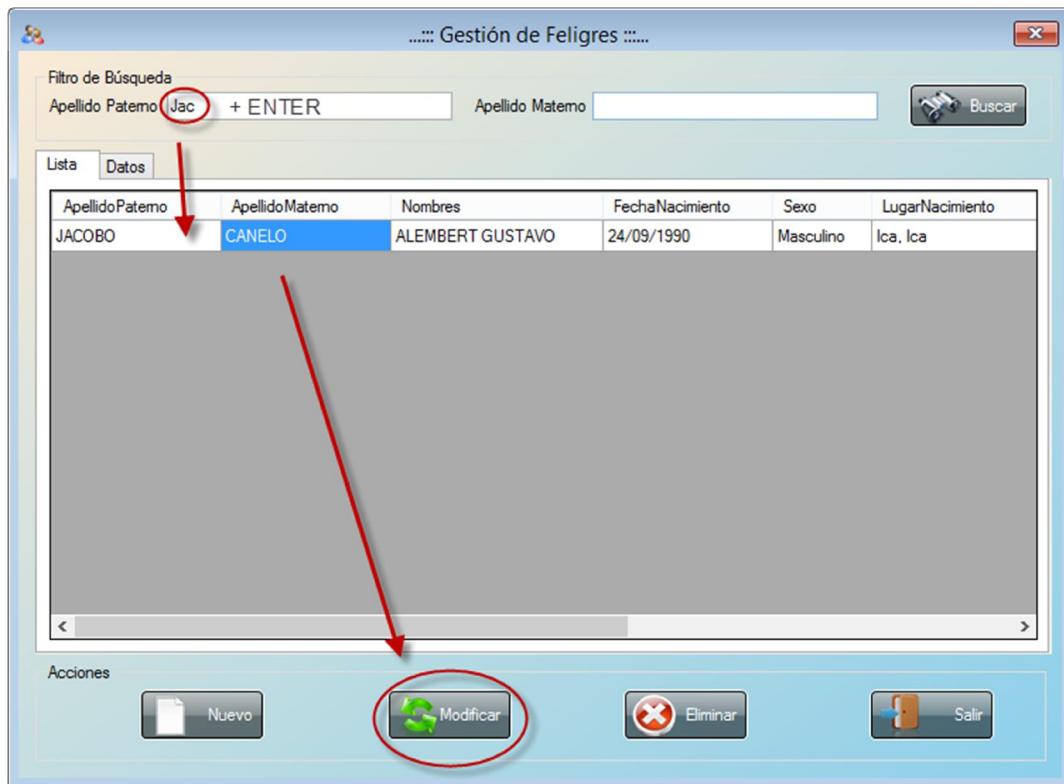
Inicialmente los registros no se cargan automáticamente, como sucedía con las ciudades o las provincias que ya aparecían cargadas al iniciar la ventana. Esta opción no está disponible en esta ventana ya que, a largo plazo, los registros que se tendrán serán inmensos y por lo tanto generarán lentitud al momento de cargar los datos.

Para buscar un registro basta con escribir las primeras letras de su apellido paterno y/o apellido materno, en la casilla correspondiente. Luego se puede hacer click en el botón “Buscar” o presionar la tecla “Enter” cuando se está en cualquiera de las 2 casillas.



## Modificar los datos de un feligrés

Para modificar los datos de un feligrés, primero debemos seleccionar al feligrés haciendo click sobre el registro y luego hacer click en el botón “Modificar”.



Una vez hecho esto, modificamos los datos que queremos cambiar y luego hacemos click en el botón “Guardar”.

<b>Datos Personales</b> Apellido Paterno: <input type="text" value="JACOBO"/> <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">12/11/1990</span> Apellido Materno: <input type="text" value="CANELO"/> Sexo: <input type="button" value="Masculino"/>	<b>Datos del Padre</b> Padre: (Nombres y Apellidos) <input type="text" value="Roberto Alonso Jacobo Hurtado"/> <input type="checkbox" value="No hay padre"/> Departamento: <input type="text" value="Ica"/> Provincia: <input type="text" value="Pisco"/> Distrito: <input type="text"/> <input checked="" type="checkbox" value="Nacional"/>
<b>Lugar de Nacimiento</b> Departamento: <input type="text" value="Ica"/> Provincia: <input type="text" value="Chincha"/> <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">12/11/1990</span> Distrito: <input type="text"/> <input checked="" type="checkbox" value="Nacional"/>	<b>Datos de la Madre</b> Madre: (Nombres y Apellidos) <input type="text" value="Rosa Angelica Canelo Canelo"/> Departamento: <input type="text" value="Ica"/> Provincia: <input type="text" value="Ica"/> Distrito: <input type="text"/> <input checked="" type="checkbox" value="Nacional"/>
<span style="border: 1px solid blue; border-radius: 50%; padding: 2px;">Guardar</span>	

ANTES:

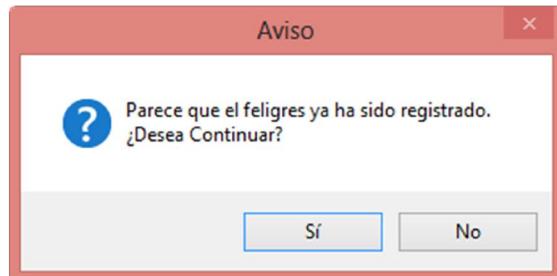
Nombres	FechaNacimiento	Sexo	LugarNacimiento	Padre
ALEMBERT GUSTAVO	24/09/1990	Masculino	Ica, Ica	Roberto Alonso Jacobo Solie

DESPUES:

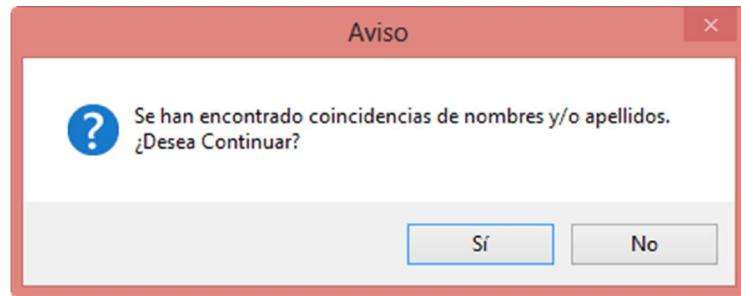
FechaNacimiento	Sexo	LugarNacimiento	Padre	Madre
12/11/1990	Masculino	Ica, Chincha	Roberto Alonso Jacobo Hurtado	Rosa Angelic

ANOTACIONES:

Si ingresamos datos que ya están registrados en otro feligrés, ya sea para agregar o modificar un feligrés, se mostrará un mensaje de advertencia. Este caso se da cuando los datos que ingresamos ya le pertenecen a otro feligrés ingresado anteriormente.



Otro caso es la “homonimia” de apellidos y nombres de una persona, es decir, 2 o más personas comparten los mismos apellidos y nombres. Si encuentra este caso el sistema mostrará el siguiente mensaje:

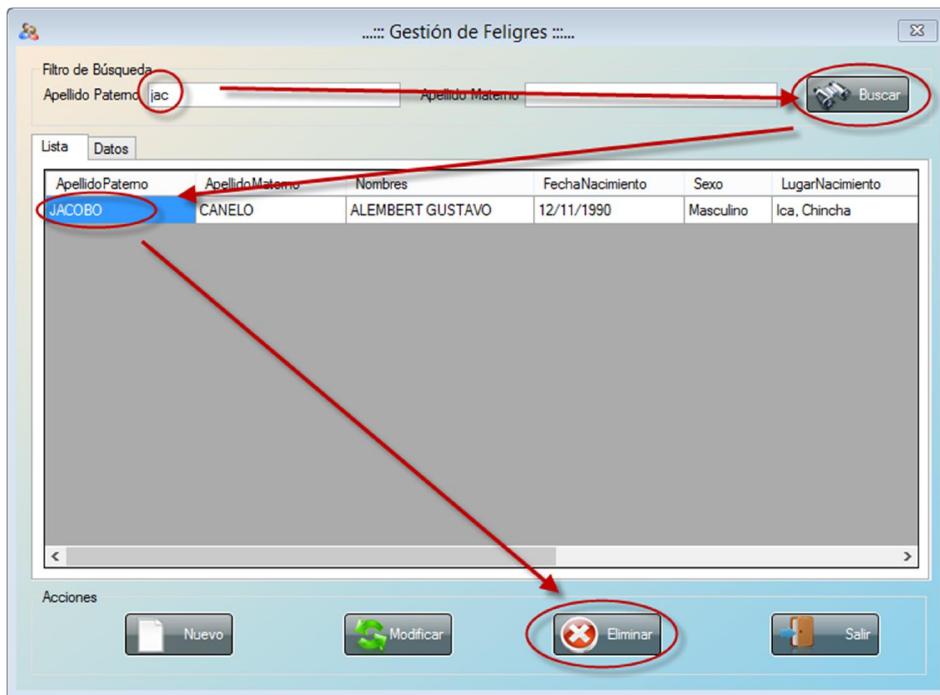


## Eliminar un feligrés

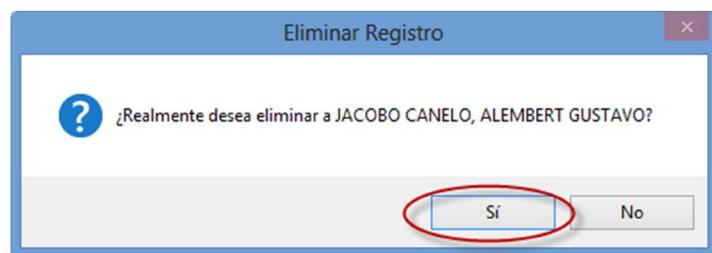
Esta función eliminará el registro en un feligrés del sistema.

OJO: Se debe tener cuidado al momento de eliminar un registro, puesto que si ese registro está siendo usado en otro proceso no podrá eliminarse, por ejemplo, si se ha registrado el bautismo del feligrés, este no podrá eliminarse.

Para eliminar un feligrés, solo debemos seleccionarlo con un click y luego hacer click en el botón "Eliminar".



Una vez hecho esto, se mostrará un mensaje de nos pedirá confirmar la eliminación del registro.



## Libro

En esta ventana se puede aperturar un nuevo libro, cerrar un libro o modificarlo y reaperturar un libro.



### Aperturar (Agregar) un Libro

Para aperturar un libro sacramental debemos hacer click en el botón “Nuevo”.

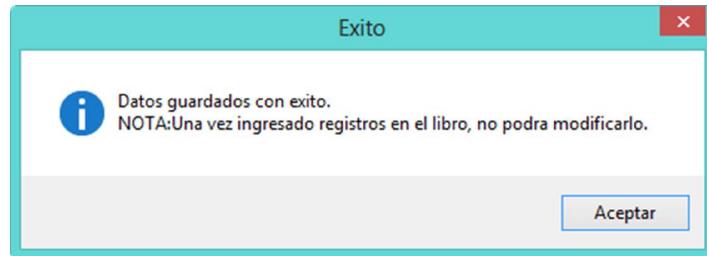
Una vez hecho esto, aparecerán unas casillas en la parte inferior para ingresar los datos.

Una vez ingresado los datos hacemos click en el botón “Guardar”.

A screenshot of a form titled "Datos" for opening a new book. The form contains several input fields: "Tipo de Libro" dropdown set to "Bautismo", "Número del Libro" input field with value "1", "Número Máx. de Folios" input field with value "500", "Cant. de Registros por Folio" input field with value "2", "Cant. de Registros" input field with value "0", "Estado" dropdown set to "Abierto", "Fecha de Apertura" date input field showing "11/11/2013", and "Fecha de Cierre" date input field which is empty. On the right side of the form is a "Guardar" button with a floppy disk icon, which is circled in red.

Antes de guardar los datos, el sistema mostrará un mensaje, preguntando si el tipo del libro es correcto ya que este no podrá modificarse más adelante. El tipo de libro se refiere a si es un libro de “Bautismo”, “Confirmación” o de “Matrimonio”.

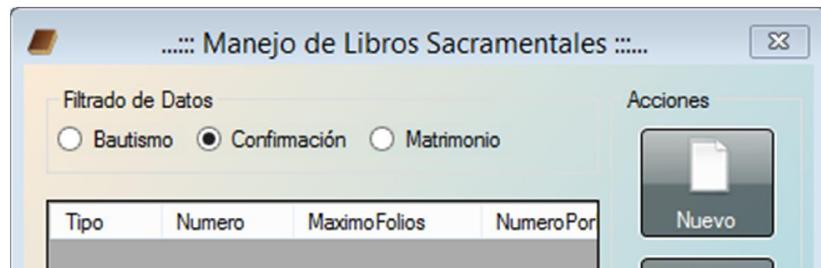




Una vez guardado el libro se mostrará un mensaje indicando el éxito de la operación y también un aviso que nos indica que una vez que ingresemos un registro en el libro, este no podrá modificarse. Es decir, si registramos un bautismo usando este libro 1, ya no podremos modificar los datos de este libro.

## Buscar un Libro

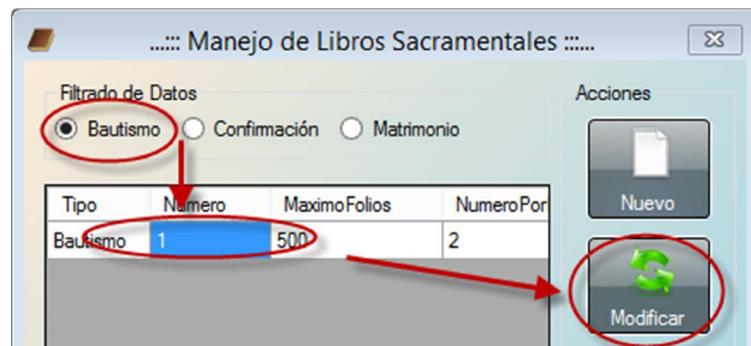
Para buscar un libro solo debemos marcar que tipo de libro es: “Bautismo”, “Confirmación” o “Matrimonio”.



## Modificar los datos de un libro

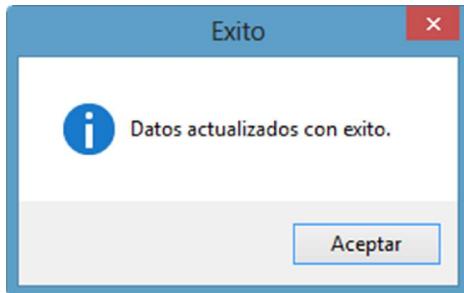
Tengamos en cuenta que si el libro que queremos modificar ya ha sido utilizado para registrar un sacramento, no podremos modificarlo.

Para modificar un Libro debemos seleccionarlo y luego hacer click en “Modificar”.



Datos		
Tipo de Libro:	Número del Libro:	Número Máx. de Folios:
Bautismo	1	500
Cant. de Registros por Folio:	Cant. de Registros:	Estado:
3	0	Abierto
Fecha de Apertura:	No se puede modificar	
11/11/2013	Fecha de Cierre:	
<input type="button" value="Guardar"/>		

Al hacer esto podremos modificar sus datos en la parte inferior. Una vez hechos los cambios hacemos click en el botón "Guardar".



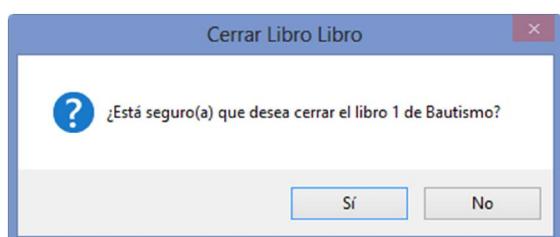
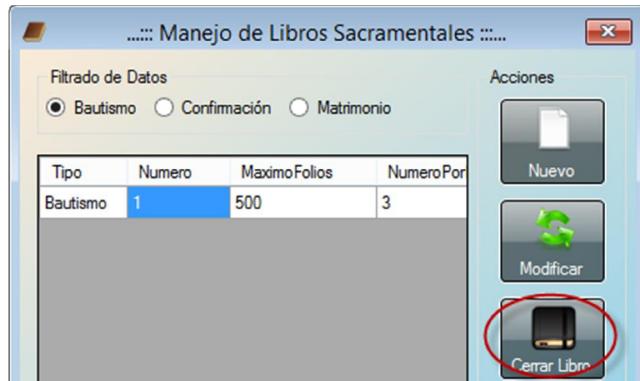
## Cerrar un Libro

Hay 2 formas en las que un libro sacramental puede cerrarse:

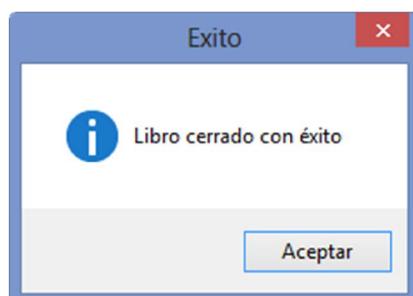
1. De forma manual.
2. Cuando el número de registros es igual al número de registros soportados.

Cabe resaltar que la segunda forma se realiza de manera automática.

Para cerrar un Libro sacramental, debemos seleccionarlo y luego hacer click en el botón "Cerrar".



Se mostrará un mensaje de confirmación para cerrar el libro seleccionado.

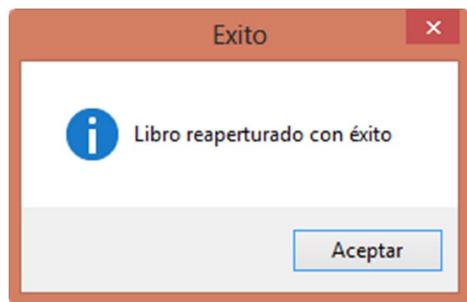
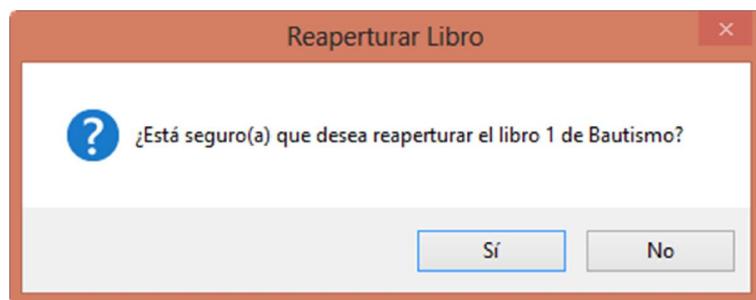


## Reaperturar un Libro

Reaperturar un libro, quiere decir volver a habilitar un libro para que guarde registros en él. No obstante, para reaperturar un libro, el número de registros que este tiene debe ser menor al número de registros soportados, es decir, si el libro soporta 500 registros como máximo y si se cerró con 498 registros, puede reaperturarse. Para reaperturar un libro solo hay que seleccionarlo y luego hacer click en el botón "Reaperturar".



Al igual que cuando cerramos un libro, también se mostrará un mensaje para confirmar la reapertura del libro.



## Templos

En esta ventana se puede agregar, modificar y/o eliminar un templo, en el cual se recibe el sacramento.

Estos templos pueden ser de 2 tipos: Capillas o Parroquias.



## Agregar un Templo

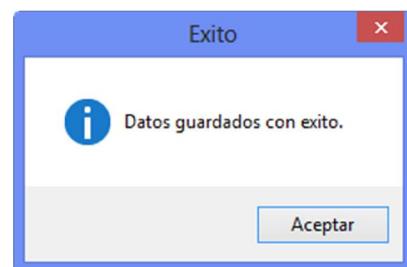
Para agregar un templo debemos hacer click en el botón “Nuevo”.

Una vez hecho, aparecerán casillas para ingresar datos en la parte inferior de la ventana.

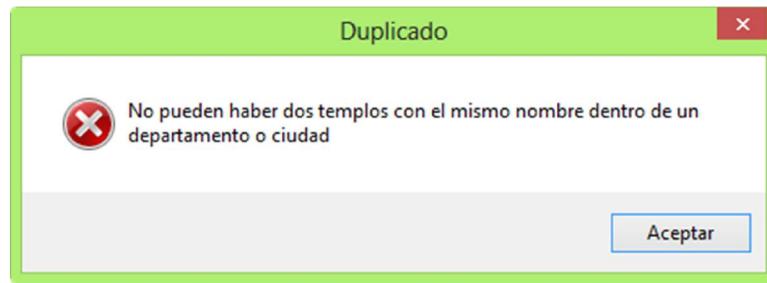
Luego de haber ingresado los datos hacer click en “Guardar”.

A screenshot of a form titled "Datos". It includes fields for "Nombre" (Nuestra Señora de Guadalupe), "Departamento" (Ica), "Provincia" (Ica), "Tipo" (Parroquia), and a checked checkbox for "Nacional". At the bottom right is a "Guardar" button, which is circled in red to indicate it is the next step.

Se mostrará un mensaje que pedirá confirmar el registro del templo en el sistema.

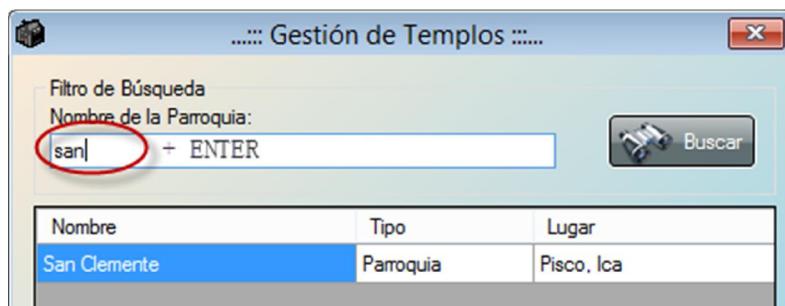
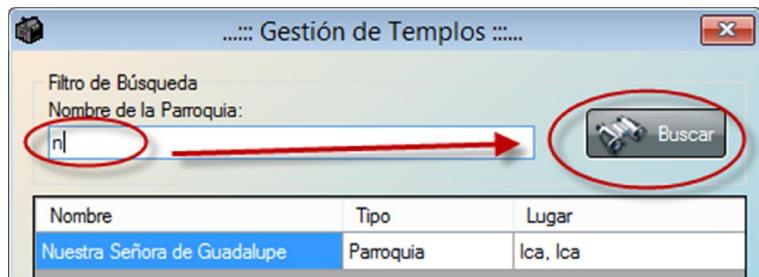


NOTA: Hay que tener en cuenta que el sistema no guardara el templo si ya existe otro templo con el mismo nombre dentro del mismo lugar.



## Buscar un Templo

Para buscar un templo, basta con escribir las primeras letras de su nombre en la casilla superior en la zona “Filtro de búsqueda” y luego presionar Enter o hacer click en el botón “Buscar”.



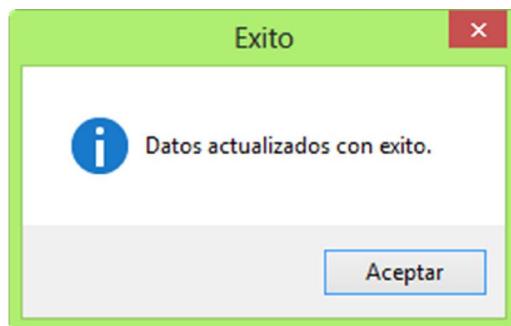
## Modificar un Templo

Para modificar los datos de un templo primero debemos seleccionarlo y luego hacer click en el botón “Modificar”.

NOTA: No se puede modificar el registro que viene por defecto en el sistema. Es decir, del templo “San Clemente de Pisco”.



Una vez hecho esto, cambiamos los datos del templo y luego hacemos click en el botón “Guardar”.



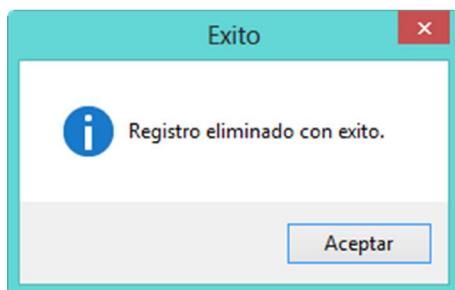
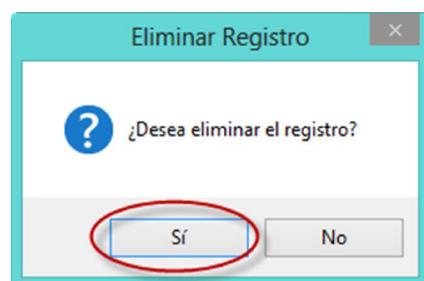
## Eliminar un Templo

Hay que tener cuidado al momento de querer eliminar un registro de algún templo. Si el templo ha sido utilizado para registrar cualquiera de los 3 sacramentos (Bautismo, Confirmación o Matrimonio) no podrá eliminarse.

NOTA: El registro por defecto del templo “San Clemente de Pisco” tampoco puede ser eliminado ni modificado.



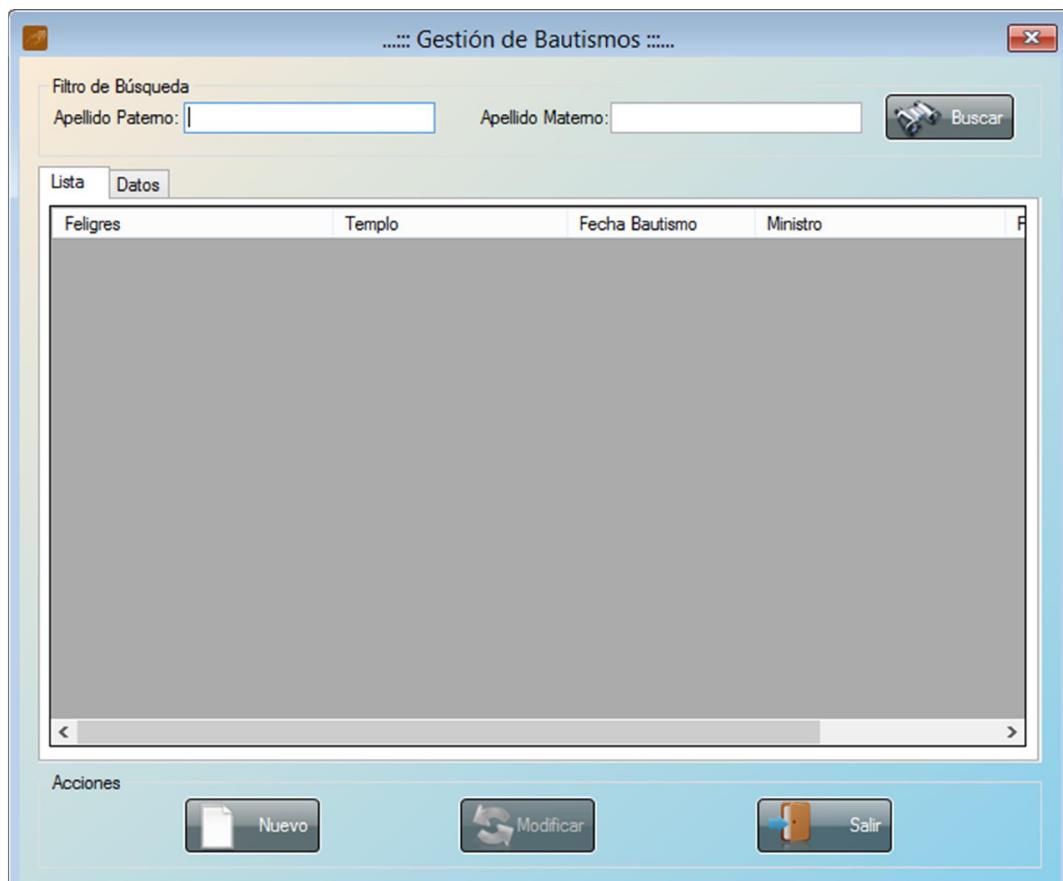
Se mostrará un mensaje que pide confirmar la eliminación del registro.



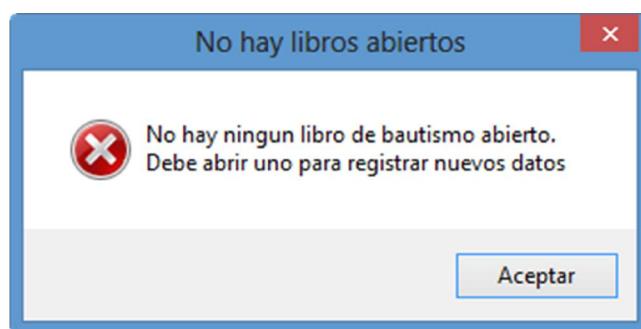
Si aceptamos, se mostrará otro mensaje indicándonos el éxito de la operación.

## Bautismo

Esta ventana sirve para registrar el bautismo de un feligrés o modificar algunos datos de un bautismo ya registrado. NO se puede eliminar un registro de bautismo.



Para poder acceder a esta ventana, primero debemos “aperturar” un libro sacramental de bautismo. En caso de no haberlo hecho o de no haber al menos 1 libro de bautismo abierto aparecerá el siguiente mensaje impidiendo entrar a la ventana.



## Registrar un Bautismo

Para registrar un bautismo, primero debemos registrar al feligrés que lo va a recibir. Esto se ha explicado anteriormente, favor de revisar nuevamente en caso de haberse olvidado.

Para poder registrar un nuevo bautismo debemos hacer click en el botón “Nuevo”.

Una vez hecho se cambiará la pestaña de “Lista” a “Datos”.

Datos del Feligrés

Apellidos y Nombres: \_\_\_\_\_ Lugar de Nacimiento: \_\_\_\_\_ Fecha Nac: \_\_\_\_\_

Padre: \_\_\_\_\_ Natural de: \_\_\_\_\_

Madre: \_\_\_\_\_ Natural de: \_\_\_\_\_

Buscar Feligrés

Datos del Bautismo

Ministro: \_\_\_\_\_ Libro: 1 Folio: 1 Número: 2

Padrino: \_\_\_\_\_ Fecha de Bautismo: 11/11/2013 Nota Marginal:

Madrina: \_\_\_\_\_

Templo: \_\_\_\_\_ Continuar

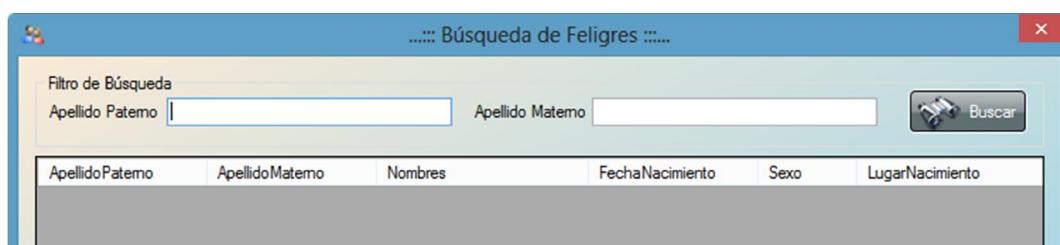
Buscar Templo

Existen 2 grandes zonas para el ingreso de datos:

1. Datos del Feligrés
2. Datos del Bautismo

Como se puede observar, las casillas de la zona “Datos del Feligrés” son grises, esto quiere decir que no podemos escribir en esas casillas.

Para poder ingresar los datos del feligrés debemos dar click en el botón “Buscar Feligrés” de la derecha.



Al hacer esto se abrirá la siguiente ventana:

Datos del Feligrés

Apellidos y Nombres: \_\_\_\_\_ Lugar de Nacimiento: \_\_\_\_\_ Fecha Nac: \_\_\_\_\_

Padre: \_\_\_\_\_ Natural de: \_\_\_\_\_

Madre: \_\_\_\_\_ Natural de: \_\_\_\_\_

Buscar Feligrés

Esta es la ventana “Búsqueda de Feligrés”. En esta ventana buscaremos al feligrés que queremos registrar en el bautismo.

Al inicio no se muestran registros.

Si queremos que se muestren “todos” los registros sólo debemos hacer click en el botón “Buscar”.

Si queremos los datos de un feligrés, debemos escribir las primeras letras de su apellido paterno y/o materno en donde corresponde y luego hacer click en el botón “Buscar”.

Como anteriormente habíamos eliminado al feligrés que se registró, se ha ingresado otro feligrés con los siguientes datos:

Apellido Paterno	Apellido Materno	Nombres	Fecha Nacimiento	Sexo	Lugar Nacimiento
NAVARRO	CARRIZALES	ROBERTO ROSINALDO	16/08/1991	Masculino	Ica, Pisco

Una vez hecho esto, seleccionamos al feligrés y luego presionamos la tecla “Enter”.

Vemos como todos los datos del feligrés seleccionado han llenado automáticamente todas las casillas.

NOTA: Los guiones consecutivos en la casilla “Padre” indican que ese feligrés no tiene un parente registrado.

Ahora que ya hemos llenado los datos del feligrés, procedemos a llenar los datos propios del bautismo. Luego para guardar los datos, hacemos click en el botón “Guardar”.

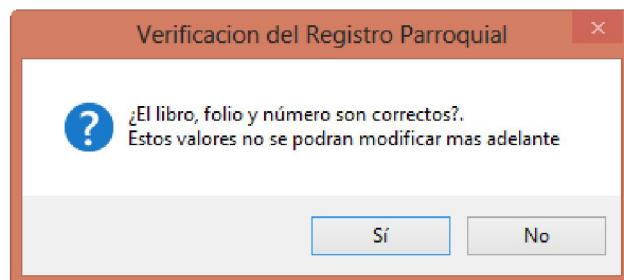
Datos del Bautismo			
Ministro:	Libro:	Folio:	Número:
P. Anselmo Becemil	1	1	1
Padrino:	Fecha de Bautismo: 02/10/1993 Nota Marginal:		
Madrina:	<input checked="" type="checkbox"/> Continuar		
Templo:	<input type="button" value="Buscar Templo"/> San Clemente de Pisco, Ica		

NOTA: Por defecto el templo donde se bautiza es en la “Parroquia San Clemente de Pisco”. En caso de no ser este el lugar donde se bautiza, debemos hacer click en el botón “Buscar Templo” para llenar los datos del templo, de la misma forma en que se hizo para el feligrés.

Al igual que se hizo con el feligrés, sólo debemos seleccionar el registro y luego presionar la tecla “Enter”.

OJO: Solo aparecerán los templos que estén en “Pisco, Ica”.

Al hacerlo, se mostrará un mensaje pidiendo la confirmación de que los datos son correctos, en especial del libro, folio y número, ya que estos datos no podrán modificarse más adelante.



NOTA: La casilla “Continuar” hace referencia a si se desea continuar la numeración del último registro, es decir, si el último registro tuvo como datos el libro 1, folio 2, número 5, esta casilla automáticamente se posicionará en el libro 1, folio 2, número 6.

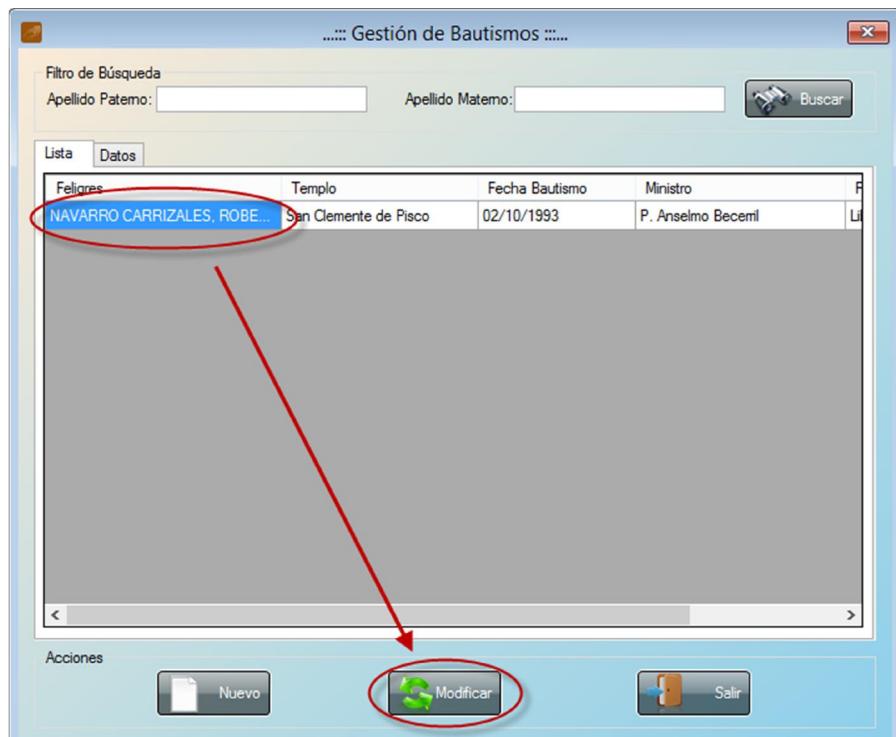
## Buscar un Bautismo

Se puede buscar un bautismo de la misma forma en que se busca un feligrés. Debemos escribir las primeras letras del apellido paterno y/o materno del feligrés en las casillas correspondientes.

.... Gestión de Bautismos ....											
Filtro de Búsqueda		<input type="button" value="Buscar"/>									
Apellido Paterno:	Apellido Materno:										
Lista    Datos <table border="1"> <thead> <tr> <th>Feligres</th> <th>Templo</th> <th>Fecha Bautismo</th> <th>Ministro</th> </tr> </thead> <tbody> <tr> <td>NAVARRO CARRIZALES, ROBE...</td> <td>San Clemente de Pisco</td> <td>02/10/1993</td> <td>P. Anselmo Becemil</td> </tr> </tbody> </table>				Feligres	Templo	Fecha Bautismo	Ministro	NAVARRO CARRIZALES, ROBE...	San Clemente de Pisco	02/10/1993	P. Anselmo Becemil
Feligres	Templo	Fecha Bautismo	Ministro								
NAVARRO CARRIZALES, ROBE...	San Clemente de Pisco	02/10/1993	P. Anselmo Becemil								

## Modificar un Bautismo

Para modificar un bautismo, primero debemos seleccionar el registro y luego hacer click en el botón “Modificar”.



Los datos encerrados en el recuadro rojo son aquellos datos que no pueden modificarse. En el caso de los sacramentos, los datos que no pueden modificarse son:

- Datos del Feligrés
- Libro, folio y número

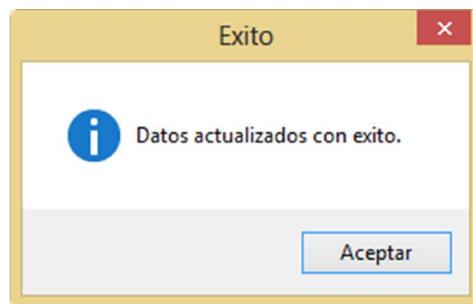
The screenshot shows the "Datos" tab of the application. The "Datos del Feligres" section is highlighted with a red box. It contains fields for "Apellidos y Nombres" (NAVARRO CARRIZALES, ROBERTO ROSINALDO), "Lugar de Nacimiento" (Ica - Pisco), "Fecha Nac." (16/08/1991), "Padre" (empty), "Natural de" (empty), and a "Buscar Feligres" button. The "Datos del Bautismo" section is also highlighted with a red box. It contains fields for "Ministro" (P. Anselmo Becemil), "Padrino" (Mario Daniel Tomes Bemaola), "Madrina" (Flor Isabel De la Cruz Prieto), "Templo" (San Clemente de Pisco, Ica), "Libro" (1), "Folio" (1), "Número" (1), "Fecha de Bautismo" (02/10/1993), "Nota Marginal" (empty), and a "Guardar" button. There is also a "Continuar" checkbox and a "Buscar Templo" button.

Hay que tener mucho cuidado al momento de registrar un bautismo.

Ahora solo debemos cambiar los datos y luego hacer click en el botón "Guardar".

Datos del Bautismo

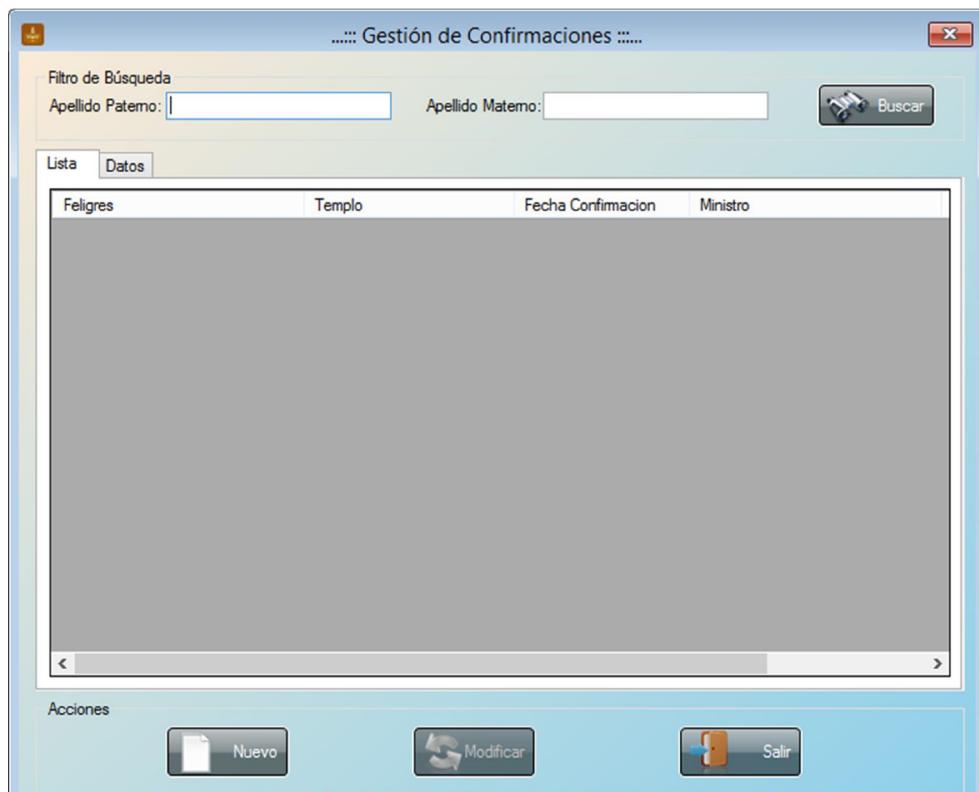
Ministro:	Libro:	Folio:	Número:	
P. Piero Sesarego	1	1	1	Guardar
Padrino:	Fecha de Bautismo:	Nota Marginal:		
Mario Daniel Torres Bemaola	02/11/1993			
Madrina:	<input checked="" type="checkbox"/> Continuar			
Flor Isabel De la Cruz Prieto				
Templo:	San Clemente de Pisco, Ica			



## Confirmación

Esta ventana permite agregar y/o modificar un registro de confirmación de un feligrés.

OJO: Al igual que en bautismo, si no hay ningún libro sacramental de confirmación abierto, no se podrá acceder a esta ventana.



## Registrar una Confirmación

Para registrar una confirmación, primero debemos registrar al feligrés que lo va a recibir. Esto se ha explicado anteriormente, favor de revisar nuevamente en caso de haberse olvidado.

Para poder registrar una nueva confirmación debemos hacer click en el botón “Nuevo”.

Una vez hecho se cambiará la pestaña de “Lista” a “Datos”.

A screenshot of a registration form divided into three sections: "Datos del Feligrés", "Datos del Bautismo", and "Datos de la Confirmación".

- Datos del Feligrés:** Fields for "Apellidos y Nombres", "Lugar de Nacimiento", and "Fecha nac.". A "Buscar Feligres" button is also present.
- Datos del Bautismo:** Fields for "Padre", "Madre", "Parroquia", "Libro", "Folio", and "Fecha de Bautismo" (set to 11/11/2013). A "Buscar Templo" button is also present.
- Datos de la Confirmación:** Fields for "Ministro", "Libro", "Folio", "Número", "Padrino o Madrina", "Templo", "Fecha de Confirmación" (set to 11/11/2013), and "Anotaciones". A "Guardar" button is located at the bottom right.

En nuestro caso tenemos 2 zonas y una zona pequeña:

1. Datos del Feligrés
  - Datos del Bautismo
2. Datos de la Confirmación

Para entender el funcionamiento de la zona “Datos del Bautismo” se agregó otro feligrés:

Datos Personales

Apellido Paterno: Torres Fecha de Nacimiento: 12/05/1993  
Apellido Materno: Avalos Sexo: Femenino  
Nombres: Patricia Alexandra

Datos del Padre

Padre: (Nombres y Apellidos) Martin Alfonso Torres Uarcaya  No hay padre  
Departamento: Lima  
Provincia: Lima  
Distrito:  Nacional

Lugar de Nacimiento

Departamento: Ica  
Provincia: Chincha  
Distrito:  Nacional

Datos de la Madre

Madre: (Nombres y Apellidos) Maria Angelica Avalos Pillaca  
Departamento: Ica  
Provincia: Pisco  
Distrito: San Andrés  Nacional

Existe como Personal  Guardar

Para llenar la primera zona “Datos del feligrés” se siguen los mismos pasos que para el bautismo. Nos vamos al botón “Buscar Feligrés” y una vez allí seleccionamos al feligrés y presionamos la tecla “Enter”.

Datos del Feligrés

Apellido y Nombres: Lugar de Nacimiento: Fecha nac:

Padre: Madre:

Datos del Bautismo

Parroquia: Fecha de Bautismo: 11/11/2013  
Libro: Folio:

Como primer caso seleccionaremos al feligrés “Navarro Carrizales”, luego presionaremos la tecla “Enter”.

..... Búsqueda de Feligres .....

Apellido Paterno	Apellido Materno	Nombres	Fecha Nacimiento	Sexo	Lugar Nacimiento
NAVARRO	CARRIZALES	ROBERTO ROSINALDO	16/08/1991	Masculino	Ica, Pisco
TORRES	AVALOS	PATRICIA ALEXANDRA	12/05/1993	Femenino	Ica, Chincha

Vemos como todos los datos de ese feligrés se han llenado de manera automática.

Datos del Feligrés		
Apellidos y Nombres:	Lugar de Nacimiento:	Fecha nac:
NAVARRO CARRIZALES, ROBERTO ROSINALDO	Ica - Pisco	16/08/1991
Padre:	Madre:	
Datos del Bautismo		
Parroquia:	San Clemente de Pisco	Buscar Templo
Libro:	1	Folio: 1
Fecha de Bautismo:	02/11/1993	

Como segundo caso vamos a seleccionar al feligrés “Torres Avalos”.

..... Búsqueda de Feligres .....

Filtro de Búsqueda	
Apellido Paterno	Apellido Materno
NAVARRO	CARRIZALES
TORRES	AVALOS
TORRES AVALOS, PATRICIA ALEXANDRA	

Datos del Feligrés		
Apellidos y Nombres:	Lugar de Nacimiento:	Fecha nac:
TORRES AVALOS, PATRICIA ALEXANDRA	Ica - Chincha	12/05/1993
Padre:	Madre:	
Martin Alfonso Torres Uarcaya	Maria Angelica Avalos Pillaca	
Datos del Bautismo		
Parroquia:	Buscar Templo	Fecha de Bautismo:
Libro:	Folio:	12/11/2013

Notamos una clara diferencia entre ambos casos. A diferencia de la primera, en la que todos los datos se llenan, en este caso los datos de la zona “Datos del Bautismo” no se han llenado y se han habilitado las casillas para ingresar datos. Esto sucede porque el sistema encontró un registro de bautismo para el feligrés “Navarro Carrizales”, y por ello los llenó automáticamente. Por el contrario, el feligrés “Torres Avalos” no tiene bautismo registrado, y como el sistema no encuentra ningún registro de bautismo, activó las casillas de dicha zona para que sean llenadas de manera automática.

Tomaremos el segundo caso e ingresaremos los datos del bautismo para el feligrés “Torres Avalos”.

Cabe resaltar que para el ingreso del templo, al hacer click en el botón “Buscar Templo”, la ventana donde buscaremos al templo mostrará a todos los templos exceptuando los de pisco, ya que si no hay registros de bautismo del feligrés, es porque no se ha bautizado en pisco.

..... Búsqueda de Templos .....

Nombre	Tipo	Lugar
Nuestra Señora del Carmen	Parroquia	Ica, Ica

Datos del Feligres

Apellidos y Nombres:	Lugar de Nacimiento:	Fecha nac:
TORRES AVALOS, PATRICIA ALEXANDRA	Ica - Chincha	12/05/1993
Padre:	Madre:	
Martin Alfonso Torres Uarcaya	Maria Angelica Avalos Pillaca	

Datos del Bautismo

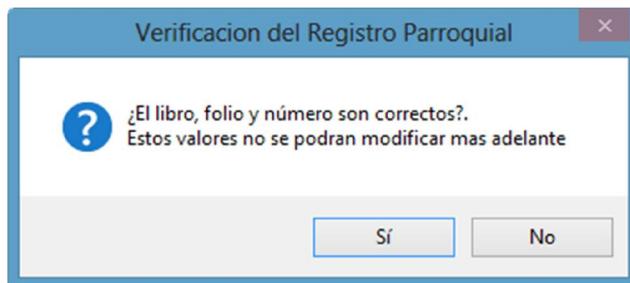
Parroquia:	Nuestra Señora del Carmen de Ica, Ica		Fecha de Bautismo:
Libro:	99	Folio:	123
			19/08/1993 

Una vez ingresado los datos de la primera zona, continuaremos con la siguiente zona que es “Datos de la Confirmación”.

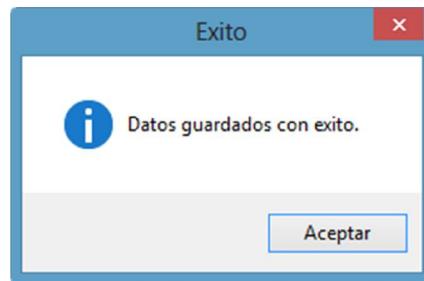
Datos de la Confirmación

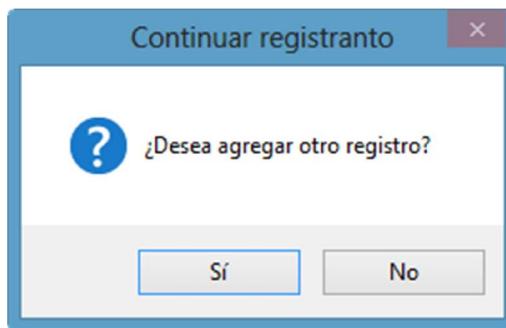
Ministro:	Libro:	Folio:	Número:	
Monseñor Hector Vera Colona	1	1	1	
Padrino o Madrina:	Fecha de Confirmación:	Anotaciones:		
Juan Carlos Sigüas Tipacti	12/11/2010 			
Templo:				
San Clemente de Pisco, Ica				

Tras terminar de ingresar los datos, hacemos click en el botón “Guardar”.



Al igual que en el bautismo, nos advierte que el libro, folio y número no se podrán modificar más adelante.





Como mayormente la confirmación es recibida por varias personas en un día, tras guardar con éxito el registro de una confirmación el sistema nos preguntará si deseamos continuar registrando confirmaciones.

Si escogemos que "Sí" se limpiarán todos los datos del feligrés y algunos del bautismo, siendo los únicos datos que no se borran: "Ministro", "Libro, Folio y Número", "Templo" y "Fecha de Confirmación".

Datos del Feligrés			
Apellidos y Nombres:	Lugar de Nacimiento:	Fecha nac:	
Padre:	Madre:	Buscar Feligres	
Datos del Bautismo			
Parroquia:	Buscar Templo		
Libro:	Folio:	Fecha de Bautismo:	11/11/2013
Datos de la Confirmación			
Ministro:	Libro:	Folio:	Número:
Monseñor Hector Vera Colona	1	1	2
Padrino o Madrina:	Fecha de Confirmación: 12/11/2013		
Templo:	Anotaciones:		
San Clemente de Pisco, Ica	Buscar Templo	Guardar	

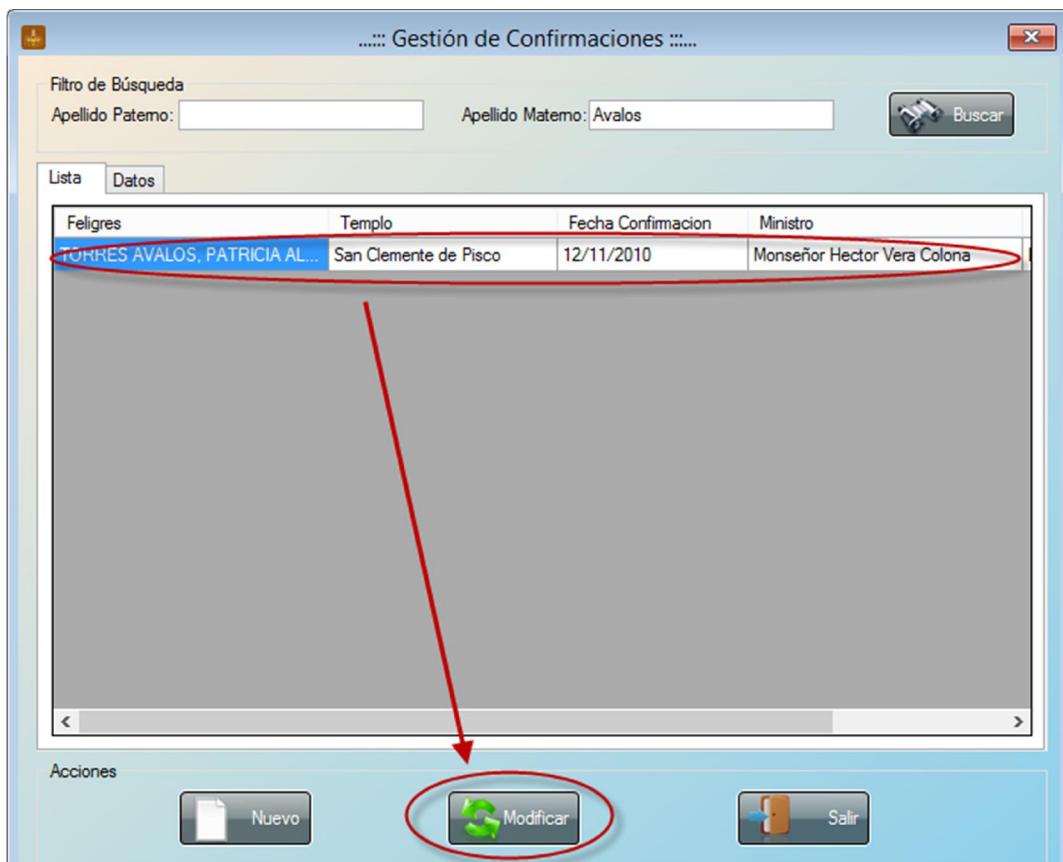
## Buscar una Confirmación

Se puede buscar una confirmación de la misma forma en que se busca un feligrés. Debemos escribir las primeras letras del apellido paterno y/o materno del feligrés en las casillas correspondientes.

..... Gestión de Confirmaciones .....											
Filtro de Búsqueda		Lista									
Apellido Paterno:	Apellido Materno:	Buscar									
Torres	Avalos										
Listado de Resultados		<table border="1"> <thead> <tr> <th>Feligres</th> <th>Templo</th> <th>Fecha Confirmacion</th> <th>Ministro</th> </tr> </thead> <tbody> <tr> <td>TORRES AVALOS, PATRICIA AL...</td> <td>San Clemente de Pisco</td> <td>12/11/2010</td> <td>Monseñor Hector Vera Colona</td> </tr> </tbody> </table>		Feligres	Templo	Fecha Confirmacion	Ministro	TORRES AVALOS, PATRICIA AL...	San Clemente de Pisco	12/11/2010	Monseñor Hector Vera Colona
Feligres	Templo	Fecha Confirmacion	Ministro								
TORRES AVALOS, PATRICIA AL...	San Clemente de Pisco	12/11/2010	Monseñor Hector Vera Colona								

## Modificar una Confirmación

Para modificar un bautismo, primero debemos seleccionar el registro y luego hacer click en el botón "Modificar".



This screenshot shows the detailed data entry screens for the selected record. The top section, 'Datos del Feligres', contains fields for Apellidos y Nombres (TORRES AVALOS, PATRICIA ALEXANDRA), Lugar de Nacimiento (Ica - Chincha), and Fecha nac. (12/05/1993). It also includes fields for Padre (Martin Alfonso Torres Uarcaya) and Madre (Maria Angelica Avalos Pillaca), with a 'Buscar Feligres' button. The middle section, 'Datos del Bautismo', contains fields for Parroquia (Nuestra Señora del Carmen de Ica, Ica), Libro (99), Folio (123), and Fecha de Bautismo (19/08/1993). The bottom section, 'Datos de la Confirmación', contains fields for Ministro (Monseñor Hector Vera Colona), Libro (1), Folio (1), Número (1), Padrino o Madrina (Juan Carlos Siguas Tipacti), Fecha de Confirmación (12/11/2010), Templo (San Clemente de Pisco), and Anotaciones. A 'Guardar' (Save) button is located at the bottom right of this section.

De la misma forma como sucede con el bautismo, los datos del feligrés y del libro, folio y número no pueden ser modificados.

La única excepción, como se puede observar, es que la zona “Datos del Bautismo” está habilitada para poder ser modificada, esto sucede porque esos datos fueron ingresados manualmente durante el registro, de haber sido llenado de forma automática estos datos no podrán ser modificados.

Una vez que hallamos cambiado los datos damos click en “Guardar”.

The screenshot shows two forms side-by-side. The top form is for baptism registration, and the bottom form is for confirmation registration. Both forms include fields for the believer's data, parents' names, baptism details, and confirmation details. The confirmation section has three fields highlighted with red boxes: 'Padrino o Madrina', 'Fecha de Confirmación', and 'Anotaciones'. Below these forms is a blue modal window titled 'Exito' with the message 'Datos actualizados con éxito.' and a 'Aceptar' button.

Datos del Feligres

Apellidos y Nombres: TORRES AVALOS, PATRICIA ALEXANDRA Lugar de Nacimiento: Ica - Chincha Fecha nac: 12/05/1993

Padre: Martín Alfonso Torres Uarcaya Madre: María Angelica Avalos Pillaca Buscar Feligres

Datos del Bautismo

Parroquia: Nuestra Señora del Carmen de Ica, Ica Libro: 99 Folio: 123 Buscar Templo Fecha de Bautismo: 19/08/1993

Datos de la Confirmación

Ministro: Monseñor Hector Vera Colona Libro: 1 Folio: 1 Número: 1 Guardar

Padrino o Madrina: Roberto Carlos Siguas Tipacti Fecha de Confirmación: 07/10/2010 Anotaciones: Primera misa de Confirmación.

Templo: San Clemente de Pisco Buscar Templo

Exito

Aceptar

Se debe notar que a diferencia del bautismo, el registro de la confirmación no sigue un patrón establecido con respecto a los números y folios, es por ello que no existe una casilla “Continuar” en esta ventana.

## Matrimonio

Esta ventana permite agregar y/o modificar un registro de matrimonio de un feligrés.

OJO: Al igual que en bautismo, si no hay ningún libro sacramental de matrimonio abierto, no se podrá acceder a esta ventana.

The screenshot shows a Windows application window titled ".... Gestión de Matrimonios ....". At the top left is a "Filtro de Búsqueda" section with four input fields: "Apellido Paterno del Esposo:", "Apellido Paterno de la Esposa:", "Apellido Materno del Esposo:", and "Apellido Materno de la Esposa:". To the right of these fields is a "Buscar" button with a magnifying glass icon. Below the search section are three tabs: "Lista" (selected), "Datos de los Esposos", and "Datos del Matrimonio". The main area is a grid table with columns: "Esposo", "Fecha Nacimiento", "Esposa", "Fecha Nacimiento", "Templo", and "Fecha". The "Esposo" column contains a placeholder "Nuevo". At the bottom left is an "Acciones" section with three buttons: "Nuevo" (with a file icon), "Modificar" (with a circular arrow icon), and "Salir" (with a door icon).

### Registrar un Matrimonio

Para registrar un matrimonio, primero debemos registrar a los feligreses que lo van a recibir. Esto se ha explicado anteriormente, favor de revisar nuevamente en caso de haberse olvidado.

Para poder registrar una nueva confirmación debemos hacer click en el botón “Nuevo”.

En este caso, tendremos 2 pestañas para el ingreso de datos:

1. Datos de los Esposos
2. Datos del Matrimonio

Dentro de la pestaña “Datos de los Esposos”, tenemos 2 zonas: “Datos del Esposo” y “Datos de la Esposa”.

Datos del Esposo		
Apellidos y Nombres:	Lugar y Fecha de Nacimiento:	Edad:
Padre:	Datos del Bautismo	Estado Civil:
Madre:	Parroquia: _____	<input type="button" value="Buscar Templo"/>
	Fecha: 11/11/2013 <input type="button" value=""/>	<input type="button" value="Buscar Esposo"/>
Datos de la Esposa		
Apellidos y Nombres:	Lugar y Fecha de Nacimiento:	Edad:
Padre:	Datos del Bautismo	Estado Civil:
Madre:	Parroquia: _____	<input type="button" value="Buscar Templo"/>
	Fecha: 11/11/2013 <input type="button" value=""/>	<input type="button" value="Buscar Esposa"/>

El llenado de los datos de los esposos es exactamente igual a como si estuviéramos llenando los datos de un feligrés en el bautismo o en la confirmación.

Hacemos click en el botón “Buscar Esposo”, luego seleccionamos el registro y presionamos “Enter”.

..... Búsqueda de Feligres .....

Filtro de Búsqueda	Apellido Paterno	Apellido Materno	Buscar		
	NAVARRO	CARRIZALES	<input type="button" value=""/>		
ApellidoPaterno	ApellidoMaterno	Nombres	FechaNacimiento	Sexo	LugarNacimiento
NAVARRO	CARRIZALES	ROBERTO ROSINALDO	16/08/1991	Masculino	Ica, Pisco

Datos del Esposo		
Apellidos y Nombres:	Lugar y Fecha de Nacimiento:	Edad:
NAVARRO CARRIZALES, ROBERTO ROSINALDO	Ica - Pisco, El día 16/08/1991	22 años
Padre:	Datos del Bautismo	Estado Civil:
_____	Parroquia: San Clemente de Pisco	<input type="button" value="Soltero"/>
Madre:	Fecha: 02/11/1993 <input type="button" value=""/>	<input type="button" value="Buscar Templo"/>
Liset Angelica Carrizales Ampuero		<input type="button" value="Buscar Esposo"/>

Como el feligrés “Navarro Carrizales” tiene un bautismo registrado, esos datos son llenados automáticamente. El único dato que hay que llenar, tenga o no registro de bautismo, es el dato “Estado Civil”.

De la misma forma llenamos los datos de la esposa.

..... Búsqueda de Feligres .....

Apellido Paterno	Apellido Materno	Nombres	Fecha Nacimiento	Sexo	Lugar Nacimiento
TORRES	AVALOS	PATRICIA ALEXANDRA	12/05/1993	Femenino	Ica, Chincha

Datos de la Esposa

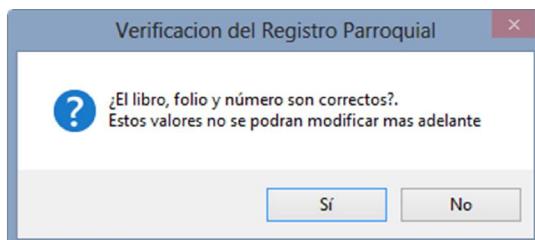
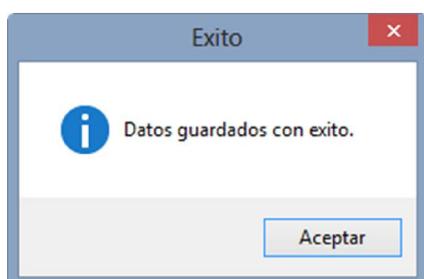
Apellidos y Nombres:	Lugar y Fecha de Nacimiento:	Edad:
TORRES AVALOS, PATRICIA ALEXANDRA	Ica - Chincha, El dia 12/05/1993	20 años
Padre:	Datos del Bautismo	Estado Civil:
Martin Alfonso Torres Uarcaya	Parroquia: Nuestra Señora del Carmen de Ica, Ica	Soltera
Madre:	Fecha: 19/08/1993	<input type="button" value="Buscar Templo"/>
Maria Angelica Avalos Pillaca		<input type="button" value="Buscar Esposa"/>

Como el feligrés "Torres Avalos" no posee un registro de bautismo en el sistema, la zona "Datos del Bautismo" se activa para que permita el ingreso de datos de forma manual. Ahora nos dirigiremos a la pestaña "Datos del Matrimonio". Procedemos a llenar los datos del matrimonio.

Sacerdote Asistente:	Templo:	
P. Mario Yepes	<input type="button" value="Buscar Templo"/>	
Padrino:	Madrina:	
Pablo Jerson Corzo Sedano	Katherine Vanesa Cordova Cespedes	
Testigo 1:	Testigo 2:	
Luis Alberto De La Cruz Prieto	Cinthya Katherine Ñañez Clemente	
Libro:	Folio:	Número:
1	1	1
Fecha:	<input type="button" value="Guardar"/>	
15/09/2012	<input checked="" type="checkbox"/> Continuar	
Anotaciones:		
<input type="text"/>		

Una vez hecho, hacemos click en el botón "Guardar".

Al igual que con el bautismo y confirmación, nos advierte que el libro, folio y número no podrán ser modificados más adelante.



## Buscar un Matrimonio

Para buscar un registro de matrimonio, podemos hacerlo escribiendo los apellidos paternos y/o maternos del esposo y/o esposa en las casillas correspondientes, y luego hacer click en el botón “Buscar”.

..... Gestión de Matrimonios .....

Filtro de Búsqueda

Apellido Paterno del Esposo: \_\_\_\_\_ Apellido Paterno de la Esposa: **Torres**

Apellido Materno del Esposo: \_\_\_\_\_ Apellido Materno de la Esposa: \_\_\_\_\_

**Buscar**

Lista Datos de los Esposos Datos del Matrimonio

Esposo	Fecha Nacimiento	Esposa	Fecha Nacimiento	Templo	Fecha
NAVARRO CARRIZALES, ...	16/08/1991	TORRES AVALOS, PATRI...	12/05/1993	San Clemente de Pisco	15/09/...

## Modificar un Matrimonio

Para modificar un matrimonio, seleccionamos el registro que queremos modificar y luego hacemos click en el botón “Modificar”.

..... Gestión de Matrimonios .....

Filtro de Búsqueda

Apellido Paterno del Esposo: \_\_\_\_\_ Apellido Paterno de la Esposa: \_\_\_\_\_

Apellido Materno del Esposo: \_\_\_\_\_ Apellido Materno de la Esposa: \_\_\_\_\_

**Buscar**

Lista Datos de los Esposos Datos del Matrimonio

Esposo	Fecha Nacimiento	Esposa	Fecha Nacimiento	Templo	Fecha
NAVARRO CARRIZALES, ...	16/08/1991	TORRES AVALOS, PATRI...	12/05/1993	San Clemente de Pisco	15/09/...

Acciones

**Modificar**

En la pestaña “Datos de los Esposos”, normalmente sólo podemos modificar el “Estado Civil” del esposo y de la esposa, sin embargo, en este caso también podemos modificar los “Datos del Bautismo” de la esposa, puesto que estos datos fueron ingresados manualmente al momento de registrar el matrimonio.

**Datos de los Esposos**

<b>Datos del Esposo</b>	<b>Lugar y Fecha de Nacimiento:</b> Ica - Pisco, El día 16/08/1991	<b>Edad:</b> 22 años
<b>Apellidos y Nombres:</b> NAVARRO CARRIZALES, ROBERTO ROSINALDO	<b>Datos del Bautismo</b>	<b>Estado Civil:</b> Soltero
<b>Padre:</b> _____	<b>Parroquia:</b> San Clemente de Pisco, Ica	<b>Buscar Templo</b>
<b>Madre:</b> Liset Angelica Carrizales Ampuero	<b>Fecha:</b> 02/11/1993	<b>Buscar Esposo</b>
<b>Datos de la Esposa</b>		
<b>Apellidos y Nombres:</b> TORRES AVALOS, PATRICIA ALEXANDRA	<b>Lugar y Fecha de Nacimiento:</b> Ica - Chincha, El día 12/05/1993	<b>Edad:</b> 20 años
<b>Padre:</b> Martin Alfonso Torres Uarcaya	<b>Datos del Bautismo</b>	<b>Estado Civil:</b> Soltera
<b>Madre:</b> Maria Angelica Avalos Pillaca	<b>Parroquia:</b> Nuestra Señora del Carmen de Ica, Ica	<b>Buscar Templo</b>
<b>Fecha:</b> 19/08/1993	<b>Buscar Esposa</b>	

En la siguiente pestaña “Datos del Matrimonio”, los únicos datos que no se pueden modificar son el libro, folio y número.

**Datos del Matrimonio**

<b>Sacerdote Asistente:</b> P. Mario Yepes	<b>Templo:</b> San Clemente de Pisco	<b>Buscar Templo</b>
<b>Padrino:</b> Pablo Jerson Corzo Sedano	<b>Madrina:</b> Katherine Vanesa Cordova Cespedes	
<b>Testigo 1:</b> Luis Alberto De La Cruz Prieto	<b>Testigo 2:</b> Cinthya Katherine Ñañez Clemente	
<b>Libro:</b> 1	<b>Folio:</b> 1	<b>Número:</b> 1
<b>Fecha:</b> 15/09/2012	<input checked="" type="checkbox"/> Continuar	<b>Guardar</b>
<b>Anotaciones:</b> _____		

Una vez hecho los cambios hacemos click en el botón “Guardar”.

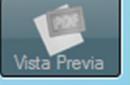
**Datos del Matrimonio**

<b>Sacerdote Asistente:</b> P. Mario Yepes	<b>Templo:</b> San Clemente de Pisco	<b>Buscar Templo</b>
<b>Padrino:</b> Pablo Jerson Corzo Sedano	<b>Madrina:</b> Katherine Vanesa Cordova Cespedes	
<b>Testigo 1:</b> Luis Rodrigo De La Cruz Prieto	<b>Testigo 2:</b> Cinthya Katherine Ñañez Clemente	
<b>Libro:</b> 1	<b>Folio:</b> 1	<b>Número:</b> 1
<b>Fecha:</b> 17/10/2012	<input checked="" type="checkbox"/> Continuar	<b>Guardar</b>
<b>Anotaciones:</b> Misa de matrimonio colectivo,		

## Actas

En esta ventana se pueden generar archivos “PDF” que luego podrán ser impresos usando las plantillas que posee la parroquia para las actas de los sacramentos.

..... Consulta Feligres .....

Datos del Feligres					
Apellidos y Nombres:	Lugar de Nacimiento:	Fecha:			
Padre:	Natural de:				
Madre:	Natural de:				
Datos del Bautismo					
Parroquia:	Libro:	Folio:	Número:	Fecha:	
Datos de la Confirmación					
Parroquia:	Libro:	Folio:	Número:	Fecha:	
Datos del Matrimonio					
Parroquia:	Libro:	Folio:	Número:	Fecha:	

Para generar un acta de cualquier sacramento, sólo debemos buscar al feligrés del cuál queremos imprimir su acta.

Para esto hacemos click en el botón “Buscar”. Luego seleccionamos al feligrés y presionamos la tecla “Enter”.

..... Búsqueda de Feligres .....

Filtro de Búsqueda

Apellido Paterno	Apellido Materno	Nombre	Fecha Nacimiento	Sexo	Lugar Nacimiento
NAVARRO	CARRIZALES	ROBERTO ROSINALDO	16/08/1991	Masculino	Ica, Pisco
TORRES	AVALOS	PATRICIA ALEXANDRA	12/05/1993	Femenino	Ica, Chincha

...:: Consulta Feligres ::...

<b>Datos del Feligrés</b>				
Apellidos y Nombres:	Lugar de Nacimiento:	Fecha:		
NAVARRO CARRIZALES, ROBERTO ROSINALDO	Ica - Pisco	16/08/1991		
Padre:	Natural de:			
Madre:	Natural de:			
Liset Angelica Carrizales Ampuero	Ica - Pisco			
<input type="button" value="Buscar"/>				
<b>Datos del Bautismo</b>				
Parroquia:	Libro:	Folio:	Número:	Fecha:
SAN CLEMENTE DE PISCO	1	1	1	02/11/1993
<input type="button" value="PDF"/> Vista Previa				
<b>Datos de la Confirmación</b>				
Parroquia:	Libro:	Folio:	Número:	Fecha:
<input type="button" value="PDF"/> Vista Previa				
<b>Datos del Matrimonio</b>				
Parroquia:	Libro:	Folio:	Número:	Fecha:
SAN CLEMENTE DE PISCO	1	1	1	17/10/2012
<input type="button" value="PDF"/> Vista Previa				

Al hacerlo automáticamente la zona “Datos del Feligrés” se llenará con los datos del feligrés. Más abajo, como se observa, las zonas “Datos del Bautismo” y “Datos del Matrimonio” se han llenado automáticamente.

La zona “Datos de la Confirmación” permanece en blanco puesto que no hay registros de confirmación de este feligrés.

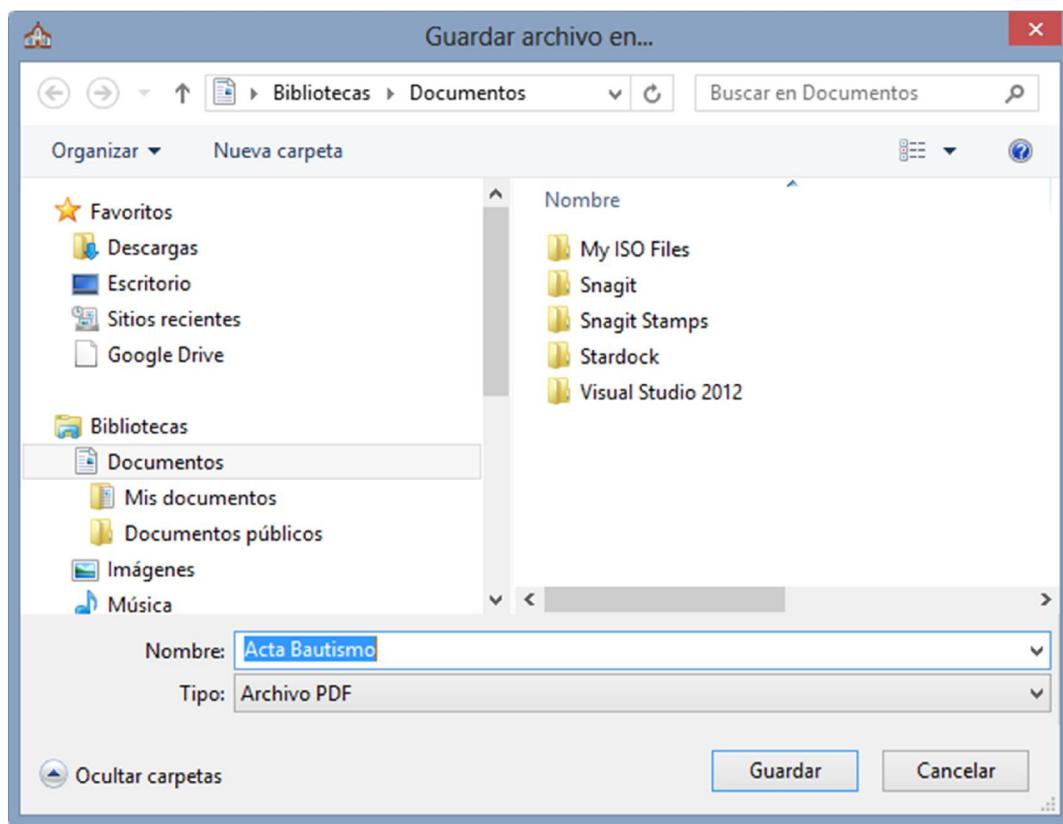
Para generar el “PDF” del acta sólo debemos hacer click en el botón “Vista Previa” del sacramento que queremos generar el acta.

En este caso escogeremos el de “Bautismo”.

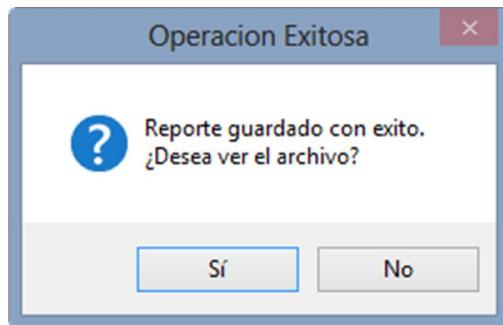
Datos del Bautismo

Parroquia:	Libro:	Folio:	Número:	Fecha:
SAN CLEMENTE DE PISCO	1	1	1	02/11/1993
<input type="button" value="PDF"/> Vista Previa				

Al hacerlo se abrirá una ventana donde podremos colocar el nombre del archivo y donde se guardará.



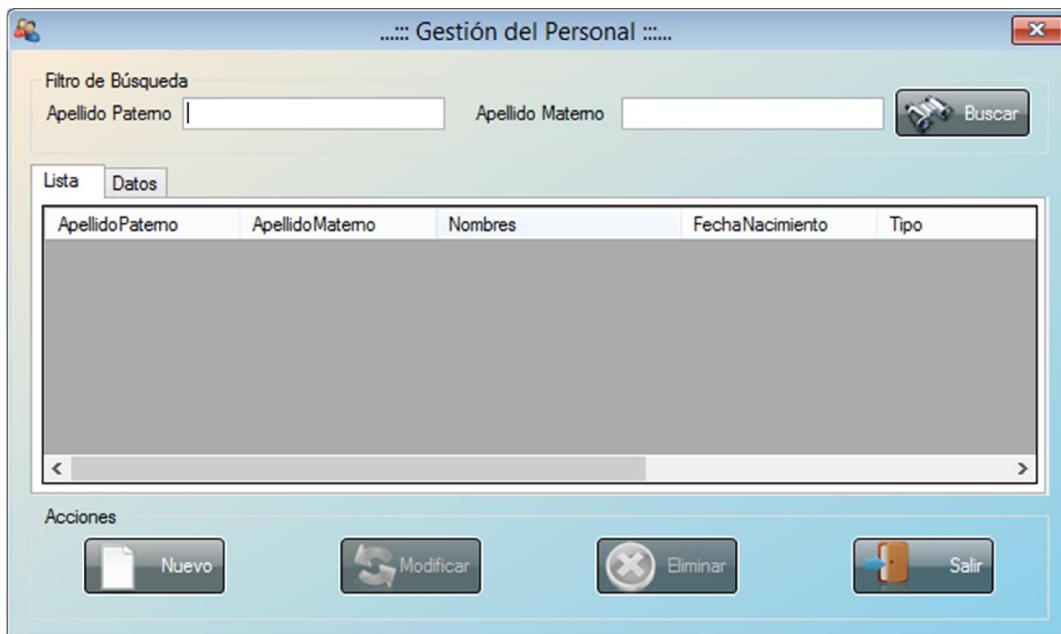
Luego de hacer click en “Guardar”, se mostrará un mensaje indicándonos que el archivo fue generado con éxito y nos pregunta si queremos ver el archivo. Este paso es opcional.



# Menú Asignación de Personal

## Personal

Esta ventana permite registrar, modificar y/o eliminar al personal con el que cuenta la parroquia.



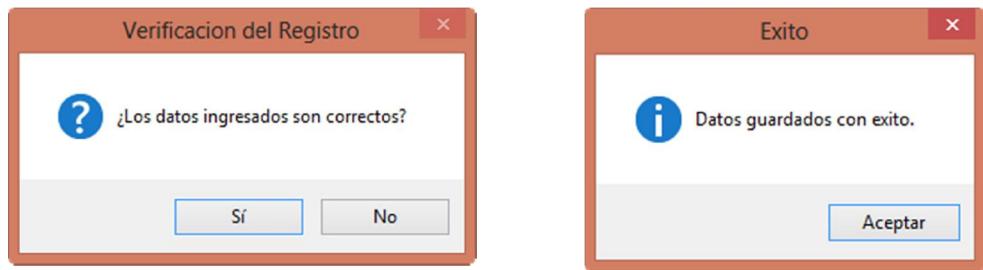
### Registrar un Personal

Para registrar a un personal, debemos hacer click en el botón “Nuevo”. Al hacerlo, se cambiará pestaña de “Lista” a “Datos”, aquí podremos ingresar los datos del personal.

This screenshot shows the "Datos Personales" and "Otros Datos" sections of the registration form. The "Datos Personales" section includes fields for "Apellido Paterno" (Soto), "Apellido Materno" (Borguez), "Nombres" (Paul Edson), "Fecha de Nacimiento" (15/07/1985), and "Sexo" (Masculino). The "Otros Datos" section includes fields for "Domicilio" (Calle Comercio 232), "Número de Teléfono" (956-456523), "Número de Celular" (956-456523), and "Tipo" (Misionero). A red arrow points from the text "Datos Opcionales" to the "Otros Datos" section, which is highlighted with a red border.

Estos datos opcionales no son necesarios llenarlos para registrar al personal.  
Una vez hecho hacemos click en el botón “Guardar”.

This screenshot shows the same "Datos Personales" and "Otros Datos" sections as the previous screenshot, but now the "Otros Datos" section is fully populated. The "Guarda" button is highlighted with a red oval.



Al igual que el feligrés, otra forma de registrar a un personal es si este ya existe como feligrés dentro del sistema, es decir, si queremos registrar como personal a un feligrés que ya está registrado, podemos hacerlo sin escribir todos sus datos nuevamente.

Seleccionamos la casilla “Existe como Feligrés” y luego hacemos click en el botón “Buscar”.

Existe como Feligrés

Datos Personales	
Apellido Paterno:	Fecha de Nacimiento:
<input type="text"/>	11/11/2013 <input type="button" value=""/>
Apellido Materno:	Sexo:
<input type="text"/>	Masculino <input type="button" value=""/>
Nombres:	<input type="button" value=""/>
<input type="text"/>	Buscar

En la nueva ventana seleccionamos al feligrés y luego presionamos la tecla “Enter”.

.....: Búsqueda de Feligres :....

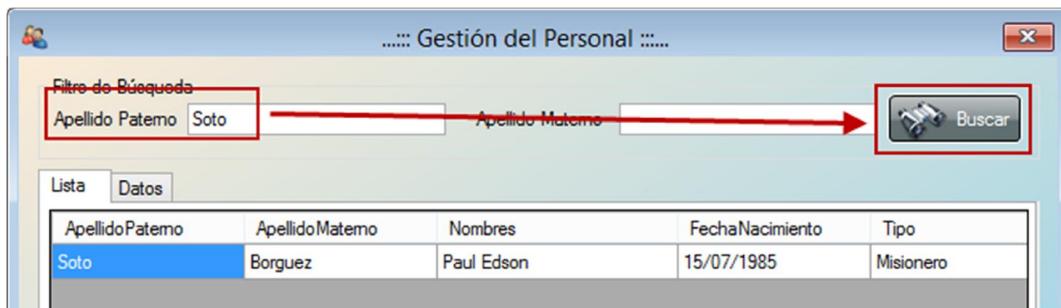
Filtro de Búsqueda

Apellido Paterno	Apellido Materno	Nombres	FechaNacimiento	Sexo	LugarNacimiento
NAVARRO	CARRIZALES	ROBERTO ROSINALDO	16/08/1991	Masculino	Ica, Pisco
TORRES	AVALOS	PATRICIA ALEXANDRA	12/05/1993	Femenino	Ica, Chincha

Datos Personales	
Apellido Paterno:	Fecha de Nacimiento:
NAVARRO	16/08/1991 <input type="button" value=""/>
Apellido Materno:	Sexo:
CARRIZALES	Masculino <input type="button" value=""/>
Nombres:	<input type="button" value=""/>
ROBERTO ROSINALDO	Buscar

## Buscar un Personal

La búsqueda del personal es exactamente igual a la búsqueda del feligrés, basta con escribir las primeras letras de su apellido paterno y/o materno en las casillas correspondientes y luego hacer click en el botón “Buscar”.



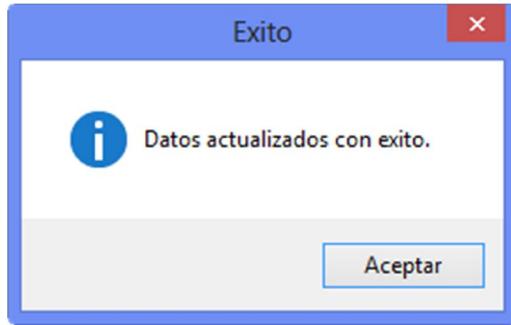
## Modificar un Personal

Para modificar un personal solo debemos seleccionarlo y luego hacer click en el botón “Modificar”.



Luego, hacemos los cambios en los datos y hacemos click en el botón “Guardar”.

Datos Personales		Otros Datos	
Apellido Paterno:	Soto	Fecha de Nacimiento:	15/06/1985
Apellido Materno:	Borguez	Sexo:	Masculino
Nombres:	Paul Edson	Domicilio:	Calle Comercio 315
		Número de Teléfono:	(056)53-4895
		Número de Celular:	956-456523
		Tipo:	Misionero
			Guardar

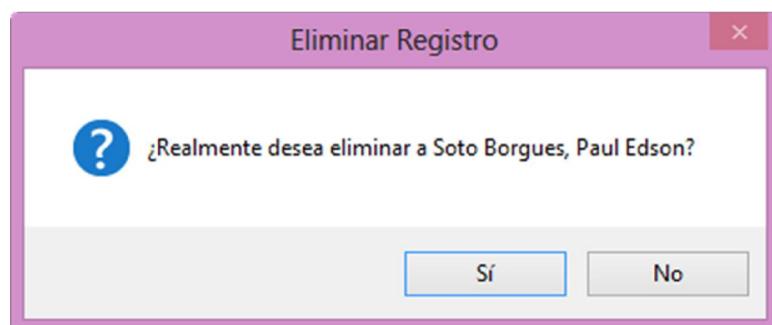


## Eliminar un Personal

Para eliminar un personal solo debemos seleccionarlo y luego hacer click en el botón "Eliminar".



Se mostrará un mensaje que nos pedirá confirmar la eliminación de ese registro.



## Comunidades

Esta ventana nos permite agregar, modificar y/o eliminar a las comunidades que hay dentro de la parroquia.

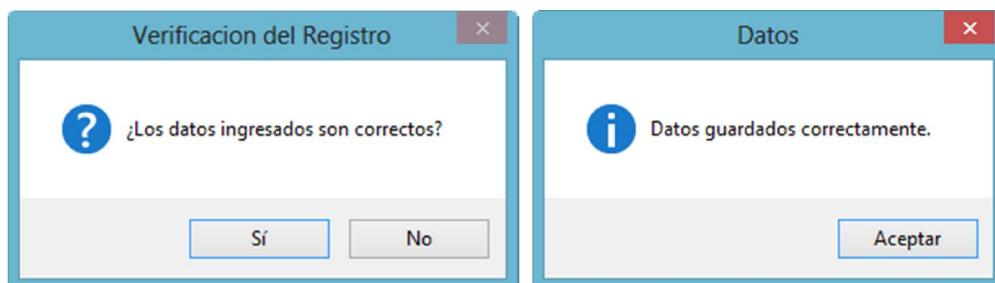


## Agregar una Comunidad

Para agregar una comunidad, primero debemos hacer click en el botón “Nuevo”. Luego llenamos los datos y hacemos click en el botón “Guardar”.

A screenshot of a "Nuevo" (New) form for adding a community. It contains fields for "Nombre" (Name) with "Cristo Joven" entered, "Tipo" (Type) set to "Grupo Parroquial", "Lugar de Reunión" (Meeting Place) with "Auditorio de Bambu" selected, "Dirección" (Address) empty, "Descripción" (Description) with the text "Grupo dedicado al canto de alabanzas al señor para las misas dominicales y de fiesta.", and a "Dia de Reunion" (Meeting Day) dropdown set to "Sábado" (Saturday). A red box highlights the "Lugar de Reunión" field. A red circle highlights the "Guardar" (Save) button, which features a disk icon.

La zona del recuadro rojo es el “Lugar de reunión” donde se reúne esta comunidad, estos lugares de pueden agregar desde la ventana “Lugar”, dentro de “Menú Básico”.



## Buscar una Comunidad

Para buscar una comunidad basta con escribir las primeras letras de su nombre en la casilla dentro de la zona “Filtro de Búsqueda” y luego hacer click en el botón “Buscar” o presionar la tecla “Enter”.



## Modificar una Comunidad

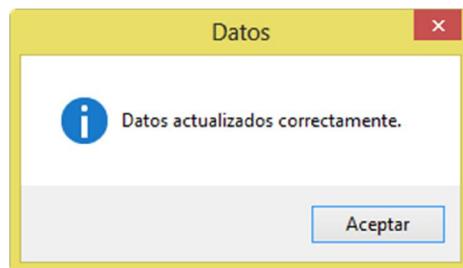
Para modificar los datos de una comunidad, primero debemos seleccionarla y luego hacer click en el botón “Modificar”.



Datos

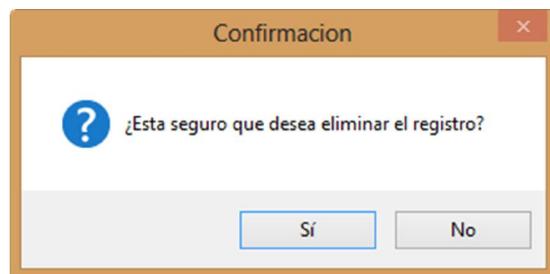
Nombre:	Tipo:
Cristo Niño	Grupo Parroquial
Lugar de Reunión:	Dirección:
Auditorio de Bambu	
Descripción:	
Grupo dedicado al canto de alabanzas al señor para las misas dominicales y de fiesta.	
Dia de Reunion:	
Sábado	
<input style="border: 1px solid red; border-radius: 50%; padding: 5px; width: 30px; height: 30px;" type="button" value="Guardar"/>	

Luego de hacer los cambios hacemos click en el botón “Guardar”.



## Eliminar una Comunidad

Para eliminar una comunidad sólo hay que seleccionar la comunidad que deseamos eliminar y luego hacer click en el botón “Eliminar”.



Se mostrará un mensaje pidiendo la confirmación para eliminar el registro.

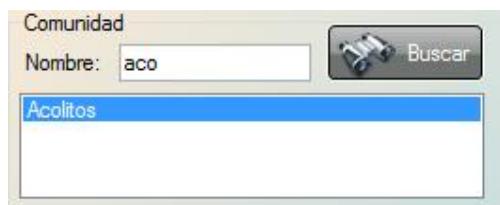
## Asignación

En esta ventana podremos asignar al personal en las distintas comunidades dentro de la parroquia. También se puede borrar la asignación hecha a una comunidad.

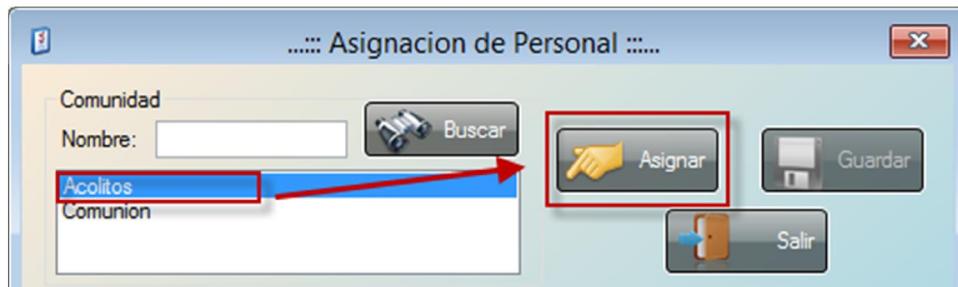


Antes de proceder con la asignación, hay que tomar en cuenta que al momento de que la ventana se abre mostrará las comunidades registradas en el sistema.

Al abrir la ventana aparecen las comunidades que hemos registrado. También podemos buscar a las comunidades colocando las primeras letras de su nombre y presionando la tecla "Enter" o haciendo click en el botón "Buscar".

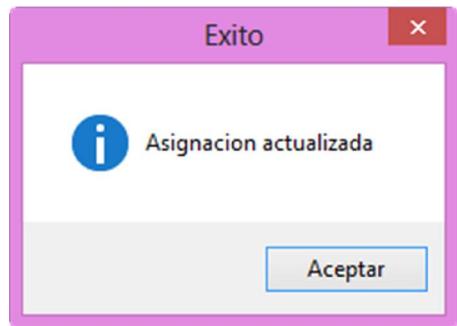


Para asignar o borrar la asignación de un personal primero seleccionamos a la comunidad y damos click en el botón "Asignar".



Al hacerlo la lista inferior izquierda donde está el personal registrado se llenará. Buscamos al personal por su apellido paterno (para buscar presionamos la tecla “Enter” luego de escribir las primeras letras de su apellido). Luego seleccionamos aquellas personas que queremos asignar y hacemos click en el botón “>”.

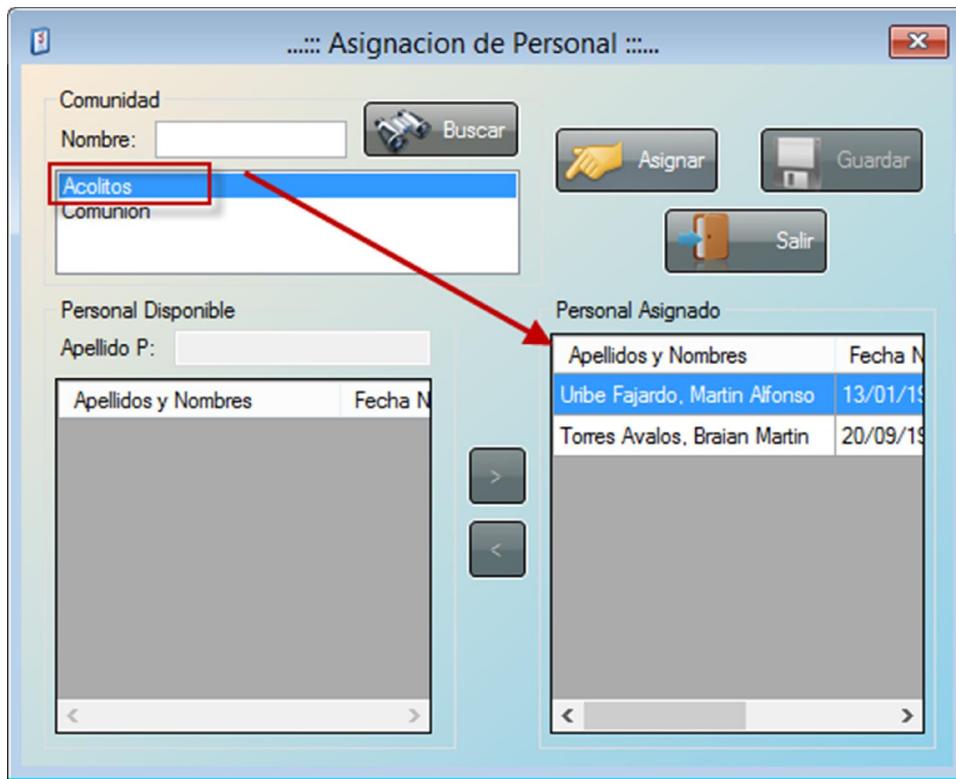




NOTA: Para quitar una asignación solo debemos hacer lo contrario, seleccionamos un personal de la lista de la derecha y hacemos click en el botón "<".

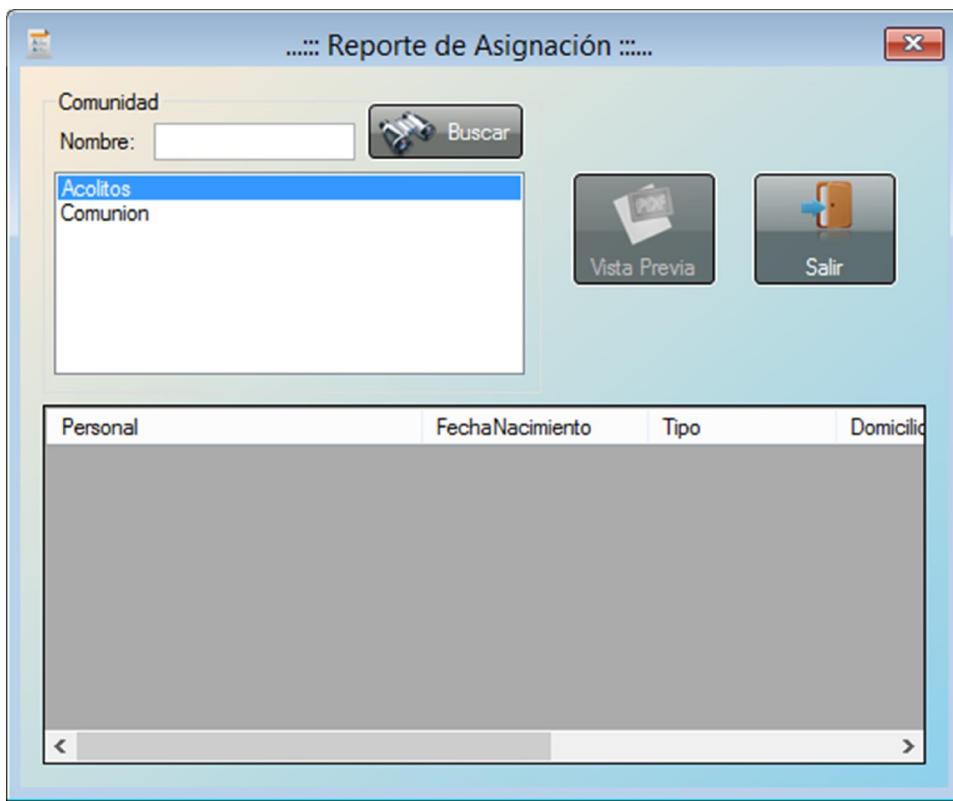
Una vez que hemos asignado al personal, hacemos click en el botón "Guardar".

Ahora, al hacer click en la comunidad, aparecerá automáticamente el personal que ya está asignado a él.



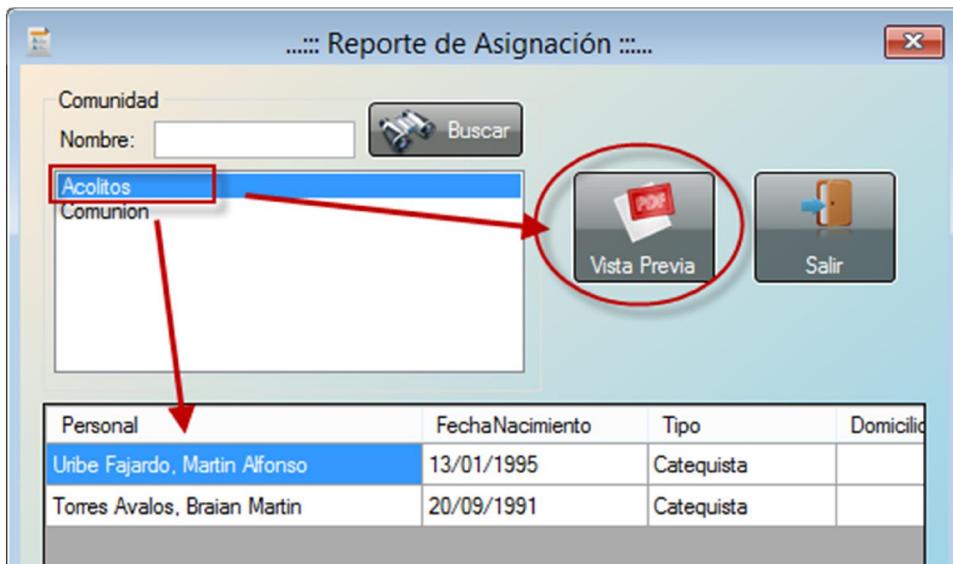
## Personal Asignado

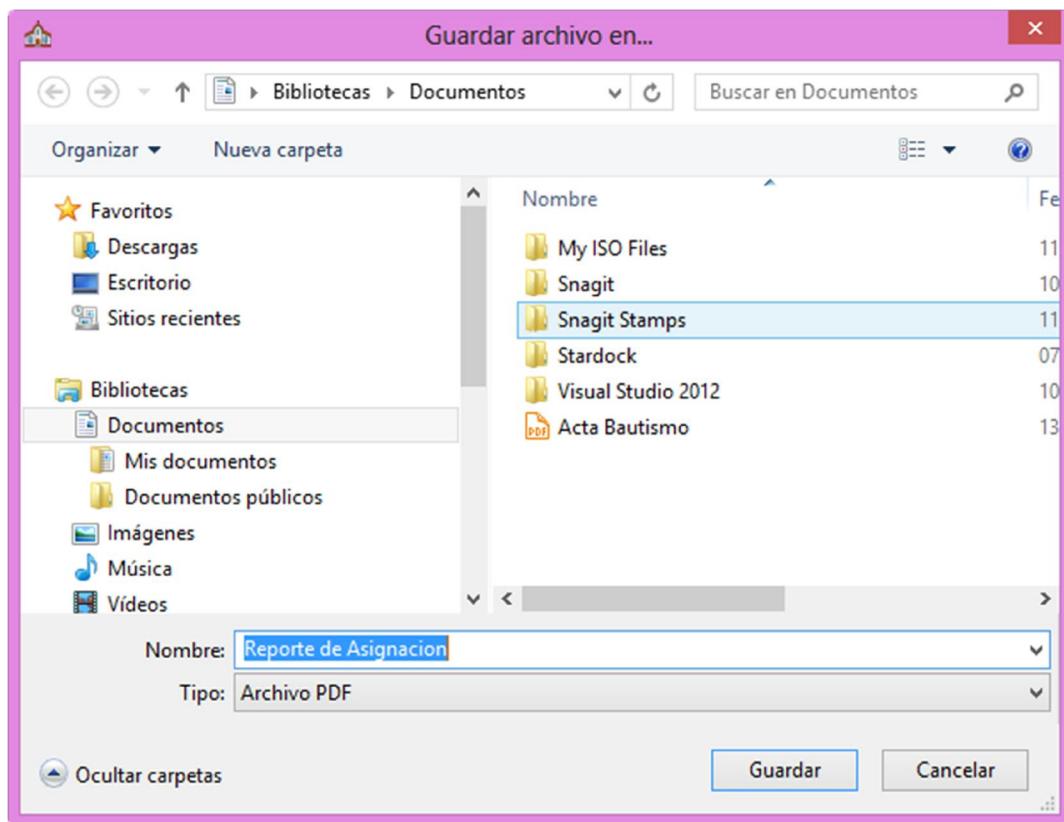
En esta ventana se pueden generar reportes en formato “PDF” sobre el personal que está asignado a una comunidad determinada.



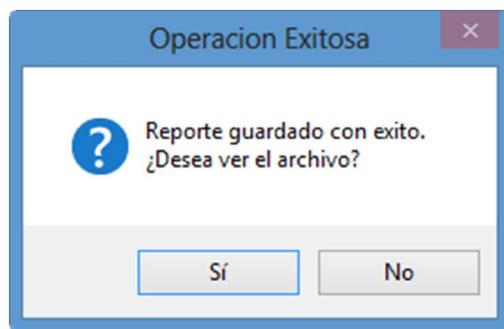
Para generar un reporte, solo debemos buscar una comunidad, escribiendo las primeras letras de su nombre y haciendo click en “Buscar”, la seleccionamos y hacemos click en el botón “Vista Previa”.

NOTA: Este botón no se activará si la comunidad no tiene personal asignado.





Aparecerá una ventana donde podremos escoger donde se guardará el archivo y el nombre que tendrá.



Una vez que el archivo se ha generado podremos tener la opción de poder visualizarlo. Esto es opcional.

### **5.3. Soporte del proyecto**

#### **5.3.1. Gestión de la configuración**

##### **Datos de Lugar y Fecha**

<b>Fecha / Hora:</b>	Lunes 18 de Noviembre del 2013 / 6:20 p.m.
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco
<b>Referencia</b>	ACTA DE AVANCES FINALES

##### **Participantes**

<b>Nombre</b>	<b>Área (Usuaria o del Equipo de Proyecto)</b>
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

##### **Objetivo del Documento**

El presente documento tiene como objetivo dar la conformidad a los avances satisfactorios del "Sistema de Control de Sacramentos y Asignación de Personal". Asimismo se indican los módulos que fueron sometidos a prueba, el estado en que se encuentran y las observaciones realizadas.

##### **Observaciones**

A continuación se detalla cada uno de los módulos.

<b>Item:</b>	<b>Módulo</b>	<b>Estado</b>	<b>Observaciones</b>
1	Feligrés	COMPLETADO	Se hizo una actualización del módulo al no pasar la primera etapa de pruebas.
2	Sacramentos	COMPLETADO	Se realizaron cambios en el código para su estabilidad.
3	Reportes de Sacramentos	COMPLETADO	Se terminó el desarrollo antes de la fecha prevista.
4	Personal	COMPLETADO	Se realizaron cambios en el código para su estabilidad.
5	Comunidades	COMPLETADO	Se realizaron cambios en el código para su estabilidad.

6	Reportes de Asignación	COMPLETADO	Se terminó el desarrollo antes de la fecha prevista.
7	Backup's	COMPLETADO	Se terminó el desarrollo antes de la fecha prevista.

### Conclusiones

Se definen como totalmente aprobados los ítems del documento, luego de haber pasado satisfactoriamente las pruebas integrales del sistema.

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

### 5.3.2. Aseguramiento de la calidad

#### Datos de Lugar y Fecha

<b>Fecha / Hora:</b>	Lunes 18 de Noviembre del 2013 / 6:20 p.m.
<b>Lugar</b>	Secretaría de la Parroquia San Clemente - Pisco
<b>Referencia</b>	ACTA DE PRUEBAS INTEGRALES

#### Participantes

Nombre	Área (Usuaria o del Equipo de Proyecto)
Juan Castillo	Gerente de Proyecto
Vicente Diez	Párroco
Luisa Cajo	Secretaria

#### Objetivo del Documento

El presente documento tiene como objetivo dar la conformidad de las pruebas realizadas al "Sistema de Control de Sacramentos y Asignación de Personal". Asimismo se indican las aprobaciones realizadas por el usuario.

#### Observaciones

A continuación se detalla cada uno de los módulos que se probaron y cuál es el estado en el que se encuentra.

Item:	Módulo	Estado	
		APROBADO	DESAPROBADO
1	Feligrés	✓	
2	Sacramentos	✓	
3	Reportes de Sacramentos	✓	
4	Personal	✓	
5	Comunidades	✓	

6	Reportes de Asignación		
7	Backup's		

### Conclusiones

Se definen como totalmente aprobados los ítems del documento marcados como "Aprobados" por los usuarios responsables. En caso que algún ítem sea marcado como "Desaprobado" se procederá a dar solución y acoplar el módulo al requerimiento del usuario, haciendo la entrega de las modificaciones del módulo a la brevedad posible.

### Aceptación Final

Para dar conformidad al presente documento, se requiere las firmas de las personas indicadas en el recuadro siguiente. En caso que este documento no contara con las firmas de los mismos o sus representantes debidamente autorizados, no se dará por aprobado el documento bajo la responsabilidad de los involucrados.

NOMBRE DEL PARTICIPANTE	FIRMA
Juan Castillo	
Vicente Diez	
Luisa Cajo	

### 5.3.3. Métricas y evaluación del desempeño

Último Periodo del 25-10-2013 al 21-11-2013

ESTADO DE AVANCE DE LOS ENTREGABLES								
FASE	ENTREGABLE			ESTADO DE AVANCE			OBSERVACIONES	
Construcción	Elaborar Manual Técnico			Actividad realizada sin problemas.				
	Elaborar Manual de Usuario			Actividad realizada sin problemas.				
Transición	Ejecutar Pruebas Integrales			Actividad realizada sin problemas.				
	Corrección de Errores Detectados			Actividad realizada sin problemas.				
	Elaborar Acta de Pruebas			Actividad realizada sin problemas.				
	Implantación del Sistema en la Parroquia			Actividad realizada sin problemas.				
	Capacitación de Usuarios			Actividad realizada sin problemas.				
	Entrega de Manuales			Actividad realizada sin problemas.				
	Elaborar Acta de Aceptación			Actividad realizada sin problemas.				
ACTIVIDADES INICIADAS EN EL PERÍODO								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	RECURSO HUMANO	FECHA INICIO	RECURSOS	
Construcción	Elaborar Manual Técnico	25-10-2013	31-10-2013	2 días	-Programador Senior -Programador Junior	25-10-2013	-Programador Senior -Programador Junior	
	Elaborar Manual de Usuario	31-10-2013	08-11-2013	2 días	-Programador Senior -Programador Junior	31-10-2013	-Programador Senior -Programador Junior	
Transición	Ejecutar Pruebas Integrales	08-11-2013	11-11-2013	1 día	-Analista de Calidad	08-11-2013	-Analista de Calidad	
	Corrección de Errores Detectados	11-11-2013	14-11-2013	1 día	-Programador Senior -Programador Junior	11-11-2013	-Programador Senior -Programador Junior	

	Elaborar Acta de Pruebas	14-11-2013	15-11-2013	3 horas	-Analista de Calidad	14-11-2013	-Analista de Calidad	
	Implantación del Sistema en la Parroquia	15-11-2013	15-11-2013	3 horas	-Jefe de Proyecto	15-11-2013	-Jefe de Proyecto	
	Capacitación de Usuarios	15-11-2013	20-11-2013	1 día	-Jefe de Proyecto -Analista Funcional	15-11-2013	-Jefe de Proyecto -Analista Funcional	
	Entrega de Manuales	20-11-2013	20-11-2013	2 horas	-Jefe de Proyecto -Analista Funcional	20-11-2013	-Jefe de Proyecto -Analista Funcional	
	Elaborar Acta de Aceptación	21-11-2013	21-11-2013	3 horas	-Analista de Calidad	21-11-2013	-Analista de Calidad	
<b>ACTIVIDADES FINALIZADAS EN EL PERÍODO</b>								
PAQUETE DE TRABAJO	NOMBRE DE ACTIVIDAD	PROGRAMADO				REAL		OBSERVACIONES
		FECHA INICIO	FECHA FIN	DURACIÓN	FECHA INICIO	FECHA FIN	RECURSOS	
Construcción	Elaborar Manual Técnico	25-10-2013	31-10-2013	2 días	-Programador Senior -Programador Junior	31-10-2013	-Programador Senior -Programador Junior	
	Elaborar Manual de Usuario	31-10-2013	08-11-2013	2 días	-Programador Senior -Programador Junior	08-11-2013	-Programador Senior -Programador Junior	

Transición	Ejecutar Pruebas Integrales	08-11-2013	11-11-2013	1 día	-Analista de Calidad	11-11-2013	-Analista de Calidad	
	Corrección de Errores Detectados	11-11-2013	14-11-2013	1 día	-Programador Senior -Programador Junior	14-11-2013	-Programador Senior -Programador Junior	
	Elaborar Acta de Pruebas	14-11-2013	15-11-2013	3 horas	-Analista de Calidad	15-11-2013	-Analista de Calidad	
	Implantación del Sistema en la Parroquia	15-11-2013	15-11-2013	3 horas	-Jefe de Proyecto	15-11-2013	-Jefe de Proyecto	
	Capacitación de Usuarios	15-11-2013	20-11-2013	1 día	-Jefe de Proyecto -Analista Funcional	20-11-2013	-Jefe de Proyecto -Analista Funcional	
	Entrega de Manuales	20-11-2013	20-11-2013	2 horas	-Jefe de Proyecto -Analista Funcional	20-11-2013	-Jefe de Proyecto -Analista Funcional	
	Elaborar Acta de Aceptación	21-11-2013	21-11-2013	3 horas	-Analista de Calidad	21-11-2013	-Analista de Calidad	
<b>MÉTRICAS DE CALIDAD EN EL PERÍODO</b>								
<b>FACTOR RELEVANTE DE CALIDAD</b>		<b>OBJETIVO DE CALIDAD</b>		<b>MEDICIÓN DE MÉTRICA</b>		<b>OBSERVACIONES</b>		
Acta de Pruebas Integrales		Obtener la aprobación de todos los elementos del módulo		FECHA		RESULTADO OBTENIDO		
Acta de Pruebas Integrales		Obtener la aprobación de todos los elementos del módulo		18-11-2013		Aprobado		
<b>CONTROLES DE CALIDAD EN EL PERÍODO</b>								
<b>ENTREGABLE</b>		<b>ESTÁNDAR DE CALIDAD APLICABLE</b>		<b>OBSERVACIONES</b>		<b>OBSERVACIONES</b>		
Sistema Completo		Revisión de funciones y fechas de entrega		FECHA		RESULTADO OBTENIDO		
Sistema Completo		Revisión de funciones y fechas de entrega		18-11-2013		Aprobado		

Tabla 49 Evaluación del Desempeño

## **CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES**

## **6.1. Conclusiones**

- Al ser la primera vez que se lleva a cabo el desarrollo de un proyecto utilizando las buenas prácticas del PMBOK, hubieron problemas notables con respecto a la administración del mismo, tanto en la parte de desarrollo del software como en la parte documental.
- En todo momento el cliente mostró se preocupación por las fechas de entrega y término del proyecto, incentivando el avance por parte del equipo de desarrollo.
- Hubieron actividades que se lograron terminar antes de las fechas acordadas, dando espacio a revisiones profundas del código fuente y optimizar las funciones del sistema.
- Los riesgos que se presentaron durante el proyecto pusieron en alerta al equipo de proyecto, pues amenazaba con modificar en gran medida los tiempos de desarrollo de actividades. Por fortuna, la eliminación de ciertos requerimientos permitió que estos no se vieran afectados.
- Se contó con apoyo externo de consulta que solventó la inexperiencia del jefe de proyecto.
- El software realizado a beneficiado a la parroquia San Clemente, en la parte interna, a la secretaría quién realizará búsquedas de registros más rápidos y exactos, y, en la parte externa, a los feligreses católicos pisqueños, que ascienden a 120 000 habitantes aproximadamente, quienes no tendrán que esperar largos periodos de tiempos para obtener las actas de los sacramentos.
- La inversión real en el desarrollo del software, el cuál fue de S/. 3, 187.00, está fundamentada en el beneficio que este ha proporcionado a la parroquia: búsquedas de registros rápidos y exactos, papeleo excesivo evitado, pérdida de la información y aumento en la calidad del servicio.

## **6.2. Recomendaciones**

- Contar con grupos de consulta externos a los cuáles se les pueda hacer preguntas sobre las distintas dudas al momento de gestionar el proyecto.
- Revisar a fondo los requerimientos y fechas de las actividades con el cliente durante la fase de planificación, para evitar así posibles cambios que puedan afectar en gran medida al proyecto.
- Contar con plantillas pre-preparadas para las pruebas del producto.
- Considerar todos los escenarios posibles para el control adecuado de los datos del negocio, revisarlos y establecer si hay ocasiones donde son necesarios datos adicionales. Esto es para prevenir la adición de ciertos campos durante la construcción del sistema.

## GLOSARIO DE TÉRMINOS

- Acta.- Relación escrita de lo sucedido, tratado o acordado en una junta.
- Catequista.- Persona que imparte catequesis (ejercicio de instruir en cosas pertenecientes a la religión).
- Concordancia .- Correspondencia o conformidad de una cosa con otra.
- Concurrencia .- Conjunto de personas que asisten a un acto o reunión.
- Contemplar.- Poner la atención en algo material o espiritual.
- Dirigir.- Guiar, mostrando o dando las señas de un camino.
- Doctrina.- Enseñanza que se da para instrucción de alguien.
- Eficacia.- Capacidad de lograr el efecto que se desea o se espera.
- Eficiencia.- Capacidad de disponer de alguien o de algo para conseguir un efecto determinado.
- Engoroso.- Difícil, molesto.
- Estimación.- Arecio y valor que se da y en que se tasa y considera algo.
- Falencia.- Engaño o error.
- Feligrés.- Persona que pertenece a determinada parroquia.
- Feligresía.- Conjunto de feligreses de una parroquia.
- Gestionar.- Hacer diligencias conducentes al logro de un negocio o de un deseo cualquiera.
- Grato.- Gustoso, agradable.
- Hincapié.- Insistir en algo que se afirma, se propone o se encarga.
- Homonimia.- Cualidad de homónimo.
- Homónimo.- Dicho de dos o más personas o cosas: Que llevan un mismo nombre.
- Idóneo.- Adecuado y apropiado para algo.
- Incidir.- Caer o incurrir en una falta, un error, un extremo.
- Interfaz.- Ventana de una determinada aplicación software.
- Iteración.- Acción y efecto de repetir.
- Mentoring.- Acción y efecto de guiar.
- Metodología.- Conjunto de métodos que se siguen en una investigación científica o en una exposición doctrinal.
- Pedagogía.- Ciencia que se ocupa de la educación y la enseñanza.
- Perjudicar.- Ocasional daño o menoscabo material o moral.
- Perjuicio.- Acción de perjudicar.
- Procedimiento.- Método de ejecutar algunas cosas.
- Proceso.- Conjunto de las fases sucesivas de un fenómeno natural o de una operación artificial.
- Propiciar.- Favorecer la ejecución de algo.
- Prototipo.- Ejemplar original o primer molde en que se fabrica una figura u otra cosa.
- Recandelerización.- Volver a calcular los tiempos en el calendario de trabajo.
- Retroalimentar.- Alimentar un sistema o circuito mediante el retorno de una parte de su salida.
- Síntesis.- Composición de un todo por la reunión de sus partes.
- Sintetizar.- Hacer síntesis.

- Software.- Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.
- Solventar.- Dar solución a un asunto difícil.
- Sponsor.- Persona o entidad que patrocina una actividad
- Status.- Posición que una persona ocupa en la sociedad o dentro de un grupo social.
- Sustentar.- Defender o sostener determinada opinión.
- Teologal.- Perteneiente o relativo a la teología.
- Teología.- Ciencia que trata de Dios y de sus atributos y perfecciones.

## BIBLIOGRAFÍA

- de la Torre Llorente, C., Zorrilla Castro, U., Calvarro Nelson, J., & Ramos Barroso, M. Á. (2010). *Guía de Arquitectura N-Capas orientada al Dominio con .NET 4.0*. España: Krasis Press.
- IBM. (2005). *Rational Unified Process Best Practices for Software Development Teams*. Lexington: Rational Software White Paper.
- Péraire, C., Edwards, M., Fernandes, A., Mancin, E., & Carroll, K. (2007). *The IBM Rational Unified Process for System Z*. New York: Redbooks.
- Project Management Institute Inc. (2008). *Guía del PMBOK 4ta Edición*. Pennsylvania: PMI Book Service Center.
- Project Management Office S.A. (n.d.). *Gestión del Valor Ganado*. Retrieved Diciembre 06, 2013, from <http://www.projectmanagementofficesa.com/biblioteca/costo/evm.asp>
- Sálas Cáceres, E. (2013). *Catecismo de la Iglesia Joven*. Lima: Editorial Salesiana.
- Turbino, M. D. (n.d.). *La Partida de Bautismo y el Estado Civil de las personas*. Retrieved Diciembre 6, 2013, from <http://revistas.pucp.edu.pe/index.php/derechopucp/article/viewFile/6485/6555>

## **FORMATOS DE LA GESTIÓN DEL PROYECTO**

**Acta de Constitución**

<b>CONTROL DE VERSIONES</b>					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0					Original
					Actualización

**ACTA DE CONSTITUCIÓN DEL PROYECTO**

<b>Nombre del proyecto:</b>		<b>Cód. Proyecto:</b>	
<b>Patrocinador del proyecto:</b>			
<b>Gerente del proyecto:</b>			
<b>Cliente del Proyecto:</b>			
<b>Preparado por:</b>			
<b>Situación actual:</b>			
<b>Propósito / Necesidad de negocio:</b>			
<b>Descripción de proyecto</b>			
<b>Descripción del producto y entregables:</b>			
<b>Criterios de aceptación:</b>			
<b>Supuestos, restricciones:</b>			
<b>Riesgos Iniciales:</b>			
<b>Objetivos de Proyecto</b>	<b>Criterios de éxito</b>	<b>Persona que aprueba</b>	
<b>Alcance</b>			
<b>Tiempo</b>			
<b>Costo</b>			
<b>Calidad</b>			

<b>Otros</b>		
<b>Resumen de Hitos:</b>		<b>Fecha esperado</b>
<b>Presupuesto estimado</b>		
<b>Autoridad del gerente de proyecto</b>		
I. Personal del proyecto		
II. Decisiones acerca de la variación del presupuesto.		
III. Decisiones técnicas		
IV. Resolución de conflictos		
V. Escalabilidad por limitación de autoridad del gerente de proyecto		
<b>Otros</b>		
<b>Aprobaciones</b>		
<hr/> Gerente del Proyecto		<hr/> Patrocinador del Py

## Declaración del Alcance

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0					Original
					Actualización

## DECLARACIÓN DEL ALCANCE

<b>Nombre de proyecto:</b>	
<b>Preparado por:</b>	
<b>Justificación del proyecto:</b>	
<b>Descripción del producto:</b>	
<b>Entregables del proyecto:</b>	
<b>Alcance:</b>	
<b>Fuera del alcance:</b>	
<b>Supuestos del proyecto:</b>	
<b>Restricciones del proyecto:</b>	
<b>Organización inicial del proyecto:</b>	
<b>Objetivos de costo (cuantificado)</b>	
<b>Objetivos de cronograma (fechas de inicio y fin)</b>	
<b>Métricas de calidad (criterios que determinarán la aceptación del proyecto)</b>	
<b>Otros objetivos</b>	
<b>Requerimientos de gestión de la configuración en el proyecto:</b>	
<b>Aprobaciones</b>	
_____ Gerente del Proyecto	_____ Patrocinador del Py

**Sistema de Control de Sacramentos y  
Asignación de Personal  
Documento - Plan Gestión de Riesgos (DPGR)  
Versión 1.0**

**Historial de revisiones**

Fecha	Versión	Descripción	Autor

**1. Introducción**

**1.1. Propósito**

**1.2. Alcance**

**2. Gestión del Riesgo**

**2.1. Identificación de Riesgos**

Listado de Riesgos, Tipo de Riesgo

ID	Descripción del Riesgo	Tipo de Riesgo
R01		
R02		

## 2.2. Análisis del Riesgo

ID	Análisis del Riesgo
R01	<b>Magnitud</b> <b>Descripción</b> <b>Impacto</b> <b>Indicadores</b>
R02	<b>Magnitud</b> <b>Descripción</b> <b>Impacto</b> <b>Indicadores</b>

## 2.3. Acciones de Prevención y de Corrección

ID	Plan de Prevención	Plan de Corrección
R01		
R02		

## 2.4. Control y Seguimiento de Riesgos

Id.	Responsable	Fecha de Terminación	Estado	Observaciones
R01				
R02				

**Responsable:** Persona o personas asignadas a la implantación de las acciones preventivas y/o correctoras

**Fecha Terminación:** Fecha límite en la cual todas las acciones anteriormente descritas deban haber sido ejecutadas por el responsable o responsables asignados.

**Estado:** Estado Actual del Riesgo y de las Acciones Preventivas y/o Correctoras.

**Observaciones:** Descripción de las observaciones encontradas para este riesgo (opcional).

### **3. Matriz de Riesgo**

Se propone la utilización de una matriz específica que sirva de soporte para la Gestión de Riesgos. Esta matriz se utilizará en las reuniones de seguimiento y/o cuando se estime necesario (en el caso de situaciones excepcionales), y su contenido será el siguiente:

<b>Id.</b>	<b>Descripción del Riesgo</b>	<b>Tipo Riesgo</b>	<b>Probab. Ocurrencia</b>	<b>Nivel de Impacto</b>	<b>Evaluación del Riesgo</b>	<b>Acciones de Prevención</b>	<b>Acción de Corrección</b>
R01							

**Id.:** Identificador de Riesgo

**Descripción del Riesgo:** Descripción Resumida del Riesgo

**Probabilidad (1 a 100):** Grado de probabilidad de que el Riesgo finalmente se produzca. Se mide en una escala de 1 a 100 (porcentual).

**Nivel de Impacto:** Grado de Impacto en el Proyecto en el caso de que el Riesgo finalmente se produjera. Se mide en una escala de 1 a 5, siendo 1=poco influyente hasta 5=fuertemente influyente.

**Probabilidad Ocurrencia:** Valor numérico resultante del producto del Grado de Probabilidad por el Grado de Impacto. Este producto dará la prioridad que tendrá la gestión de este Riesgo y la implantación de sus medidas preventivas o correctoras.

**Acciones Prevención:** Descripción de las Acciones o Medidas a Adoptar para evitar (mitigar) la aparición final del Riesgo

**Acciones Corrección:** Descripción de las Acciones o Medidas a Adoptar en el caso en el que el Riesgo finalmente se haya producido

APROBACIONES	
Gerente del Proyecto	Patrocinador del Py

## Plan de Gestión de Adquisiciones

Matriz de Adquisiciones

NOMBRE DEL PROYECTO: SISTEMA DE CONTROL DE SACRAMENTOS Y ASGINACION DE PERSONAL											
CTO O SERVICIO A ADQUIRIR	CÓDIGO DE ELEMENTO WBS	TIPO DE CONTRATO	DOCUMENTOS DE LA ADQUISICIÓN	REQUERIMIENTO DE ESTIMACIONES INDEPENDIENTES	ÁREA/ROL/PERSONA RESPONSABLE DE LA COMPRA	MANEJO DE MÚLTIPLES PROVEEDORES	PROVEEDORES PRE-CALIFICADOS	CRONOGRAMA			
								Planif. ADQUIS	Efectuar ADQUIS	Admin. ADQUIS	Cerrar ADQUIS
								Del al	Del al	Del al	Del al

<b>ADQ01</b>		
<b>Nombre del proyecto:</b>		
<b>Código de proyecto:</b>	<b>Tipo de proyecto:</b>	<b>Cliente:</b>
<b>Elaborado por:</b>	<b>Aprobado por:</b>	
<b>ENUNCIADO DEL TRABAJO</b>		
Nombre del Proveedor:		
<b>Descripción:</b>		
<b>Parámetros de Costo:</b>		
<ul style="list-style-type: none"> <li>▪ .....S/. 00.00</li> </ul>		

<b>Aprobaciones</b>	
Gerente del Proyecto	Patrocinador del Py

## Plan de Gestión de Calidad

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0					Original
					Actualización

### Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal

#### Elaboración del Plan de Gestión de Calidad

PLAN DE GESTIÓN DE CALIDAD		Fecha de Aprobación:
Descripción del Sistema de Calidad para el Proyecto		
Estructura organizacional de la Calidad		
Roles y Responsabilidades		
		-
Procedimientos y Normas		
Procesos		
-		
Recursos		
-		
Actividades de Calidad		
Type	Fecha	Descripción
Elaborado por	Aprobado por	

## Plan de Gestión de Recursos Humanos

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0					Original

### PLAN DE GESTIÓN DE RECURSOS HUMANOS

Nombre del Proyecto	Siglas del Proyecto

COMPONENTE	DESCRIPCIÓN					
Jefe del Proyecto:						
<b>ORGANIGRAMA DEL PROYECTO</b>						
Ver Organigrama del Proyecto (EDT) – versión 1.0						
<b>ROLES Y RESPONSABILIDAD</b>						
Ver Matriz de Asignaciones de Responsabilidades (RAM) – versión 1.0						
<b>Requerimiento de Personal</b>		<b>Descripción de Roles</b>				
(D): Dirige (P): Participa (V): Valida		-				
<b>OBJETIVO DEL PLAN</b>						
<b>REQUERIMIENTO DE PERSONAL</b>						
<b>CRITERIOS DE LIBERACIÓN DEL PERSONAL DEL PROYECTO:</b>						
ROL	CRITERIO DE LIBERACIÓN	¿CÓMO?	DESTINO DE ASIGNACIÓN			
<b>CAPACITACIÓN, ENTRENAMIENTO, MENTORING REQUERIDO</b>						
<b>SISTEMA DE RECONOCIMIENTOS Y RECOMPENSA</b>						
<b>CUMPLIMIENTO DE REGULACIONES, PACTOS Y POLÍTICAS</b>						
<b>REQUERIMIENTO DE SEGURIDAD</b>						
<b>APROBACIONES</b>						
Gerente del Proyecto		Patrocinador del Py				

**Plan de Gestión de Comunicaciones**

***PLAN DE PROYECTO***

**PLAN DE GESTIÓN DE LA COMUNICACIÓN**

**[Fecha]**

Definición del Esquema de Comunicaciones

Tipos de Comunicación

Esquema de Comunicación de Documentos

Documento	Tipo	Usuario Destino	Medio de Entrega	Formato	Frecuencia	Responsable

Estándares para las Reuniones

Estándares para los Archivos

Almacenamiento de Documentación del Proyecto

[Plantillas de Comunicación]

APROBACIONES	
_____ Gerente del Proyecto	_____ Patrocinador del Py

## **ANEXOS**

## Anexo Nº 1

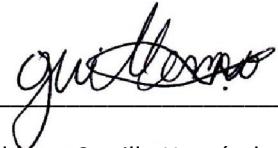
CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	JC	LL	VD	04/06/2013	Original
2.0	JC	LL	VD	30/09/2013	Actualización

## DECLARACIÓN DEL ALCANCE

<b>Nombre de proyecto:</b> Implementación de un Sistema Informático de Gestión para el Control de Sacramentos y Asignación de Personal	
<b>Preparado por:</b> Juan Guillermo Castillo Hernández	
<b>Justificación del proyecto:</b>	<p>La parroquia requiere mayor eficiencia y control de los expedición de actas de sacramentos y la asignación de su personal, para mejorar la calidad de servicio a la feligresía debido a que:</p> <ul style="list-style-type: none"> <li>• La feligresía está poco satisfecha respecto al tiempo de respuesta a la atención de la expedición de actas.</li> <li>• Existe demora en responder a las solicitudes de la expedición de actas de la feligresía.</li> <li>• Falta de control en asignación de personal a las distintas comunidades de la parroquia, desconociendo en algunos casos quienes forman parte de alguna comunidad.</li> </ul> <p>Por lo que se obtendrá información de la parroquia, del registro de sacramentos, del personal, de las comunidades y eventos. El administrador tendrá los datos vitales en relación con la parroquia, y una vista instantánea de la información de los registros de los sacramentos y el modo en que asigna a su personal, lo que se traducirá en una mayor rentabilidad debido a una mayor capacidad para atender a la feligresía y la disminución de los costos operativos.</p>
<b>Descripción del producto:</b>	El producto a entregar es un Sistema de Control de Sacramentos y Asignación de Personal, el cual controlará el registro de los sacramentos y la asignación de personal, aplicativo que permita al cliente de la Parroquia ingresar los registros de los sacramentos y poder hacer una asignación más eficiente de su personal. Por el lado de la Parroquia permitirá la gestión de dichos requerimientos, haciendo posible el control de los sacramentos y la asignación del Personal.
<b>Entregables del proyecto:</b>	<p><b>Principales Entregables del Producto</b></p> <ul style="list-style-type: none"> <li>• Sistema Control de Sacramentos y Asignación de Personal</li> <li>• Módulo de Feligrés</li> </ul>

	<ul style="list-style-type: none"> <li>• Módulo de Sacramentos</li> <li>• Módulo de Reportes de Sacramentos             <ul style="list-style-type: none"> <li>- Sacramentos recibidos por feligrés</li> </ul> </li> <li>• Módulo de Personal</li> <li>• Módulo de Comunidades/Grupos</li> <li>• Módulo de Reportes de Asignación</li> <li>• Módulo de Backups</li> <li>• Manual técnico</li> <li>• Manual de usuario</li> <li>• Informe de Pruebas integrales</li> </ul> <p><b>Gestión del Proyecto</b></p> <ul style="list-style-type: none"> <li>• Acta de Constitución del Proyecto (Project Charter)</li> <li>• Plan de Gestión del Proyecto</li> <li>• Plan de Gestión del Alcance</li> <li>• Estructura del desglose del trabajo - EDT (WBS)</li> <li>• Cronograma del proyecto</li> <li>• Plan de gestión de costos del proyecto</li> <li>• Plan de Gestión de Requerimientos</li> <li>• Plan de Gestión de Cambios</li> <li>• Plan de Gestión de la Calidad</li> <li>• Plan de Gestión de riesgos</li> <li>• Plan de Gestión de las comunicaciones</li> <li>• Plan de Gestión de adquisiciones</li> <li>• Plan de Gestión de RRHH</li> <li>• Lecciones aprendidas</li> <li>• Aceptación Final del Proyecto</li> </ul>
<b>Alcance:</b>	<p>Desarrollar un Sistema de control de sacramentos y asignación de personal.</p> <ul style="list-style-type: none"> <li>• Registrar datos específicos del feligrés.</li> <li>• Registrar datos específicos de cada sacramento.</li> <li>• Permitir la impresión de actas de sacramentos.</li> <li>• Registrar datos específicos del personal.</li> <li>• Registrar datos específicos de las comunidades.</li> <li>• Llevar el control de la asignación de personal a las diferentes comunidades.</li> <li>• Permitir la impresión de reportes de asignación.</li> </ul>
<b>Fuera del alcance:</b>	<ul style="list-style-type: none"> <li>- No se incluirá la mejora de procesos y/o eliminación de actividades que no justifican esfuerzo o no agregan valor.</li> </ul>
<b>Supuestos del proyecto:</b>	<ul style="list-style-type: none"> <li>- Se cuenta con plataforma tecnológica en la Parroquia San Clemente de Pisco requerida para el proyecto: Computadora de Escritorio e impresora.</li> </ul>

	<ul style="list-style-type: none"> <li>- El responsable del proyecto estará comprometido con la ejecución del proyecto.</li> </ul>
<b>Restricciones del proyecto:</b>	<ul style="list-style-type: none"> <li>- El costo del proyecto no debe sobrepasar del monto base fijado de S/. 13,065.80</li> <li>- El plazo del desarrollo del proyecto debe estar dentro de los 8 meses planteados para la entrega del Proyecto, sin oportunidad a ampliar el plazo fijado.</li> <li>- Al siguiente día de cumplido el plazo de desarrollo el Proyecto, este ya debe encontrarse instalado y ejecutándose correctamente dentro de la organización.</li> <li>- Toda la documentación generada durante la ejecución del proyecto deberá emplear como base los formatos de la parroquia.</li> </ul>
<b>Organización inicial del proyecto:</b>	<ul style="list-style-type: none"> <li>-Párroco</li> <li>-Secretaria</li> <li>- Gerente de Proyecto</li> </ul>
<b>Objetivos de costo (cuantificado)</b>	El costo del Proyecto está fijado en S/. 13,065.80 (Trece mil sesenta y cinco nuevos soles con ochenta céntimos), monto que será asumido por el Responsable o Gerente del Proyecto.
<b>Objetivos de cronograma (fechas de inicio y fin)</b>	<p>Inicio: 19/04/2013  Fin: 21/11/2013</p>
<b>Métricas de calidad (criterios que determinarán la aceptación del proyecto)</b>	Se aplicará el ciclo de vida de desarrollo de software (SDLC). Se evaluará la calidad del Producto de Software validando con el cliente las funciones del sistema, con los requerimientos que este debe cumplir.
<b>Otros objetivos</b>	
<b>Requerimientos de gestión de la configuración en el proyecto:</b>	<p>La estabilidad del alcance del proyecto se operará de la siguiente forma:</p> <ul style="list-style-type: none"> <li>• Se realizará reuniones quincenales o mensuales para el control del alcance del proyecto que será realizado por el Gerente del proyecto, sponsor y la secretaria.</li> <li>• El párroco es el Sponsor del proyecto, por tanto es quien podrá solicitar un cambio en el alcance del proyecto, el cual será canalizado por el Gerente del Proyecto utilizando la plantilla de solicitud de cambios en el alcance.</li> <li>• El Gerente de Proyecto informará al Comité de Seguimiento sobre el impacto en el proyecto de las variaciones en el alcance, debido a los cambios solicitados. El impacto se informará en términos de variación del plazo, presupuesto o alcance.</li> </ul>

	<ul style="list-style-type: none"> <li>• El Comité de Seguimiento del Proyecto se reunirá para analizar el impacto de los cambios en el proyecto y será el responsable de aprobar o rechazar los cambios solicitados.</li> <li>• Las acciones correctivas serán generadas por el analista del proyecto, y serán revisadas y aprobadas por el Gerente del Proyecto.</li> </ul> <p>Los cambios del alcance serán identificados:</p> <ul style="list-style-type: none"> <li>• A partir de los cambios solicitados por los usuarios líderes: Gerente general y Gerente de producto.</li> <li>• Durante la elaboración y/o revisión del EDT.</li> <li>• Por el control y/o seguimiento de Plan del Proyecto.</li> </ul> <p>Los cambios del alcance serán clasificados por medio de su impacto:</p> <ul style="list-style-type: none"> <li>• Alto: Cuando el cambio del alcance del proyecto cambia en un 50% el objetivo del proyecto.</li> <li>• Mediano: Cuando el cambio del alcance del proyecto cambia en un 30% el objetivo del proyecto.</li> <li>• Bajo: Cuando el cambio del alcance del proyecto cambia en un 10% el objetivo del proyecto.</li> </ul>
<b>Aprobaciones</b>	
 <hr/> <p>Juan Guillermo Castillo Hernández Gerente del Proyecto 04/10/2013</p>	 <hr/> <p>Vicente Diez Varona Párroco Patrocinador del Py 04/10/2013</p>

**Anexo Nº 2**

**Sistema de Control de Sacramentos y  
Asignación de Personal  
Documento - Plan Gestión de Riesgos (DPGR)  
Versión 1.0**

**Historial de revisiones**

Fecha	Versión	Descripción	Autor
28/06/2013	1.0	Documento de Identificación y Análisis de Riesgos	Juan Castillo

## 1. Introducción

Uno de los elementos clave a la hora de asegurar el éxito en el proyecto, medido en términos de cumplimiento de plazos, costes, alcance funcional y calidad final de la solución, es la Gestión de Riesgos. Implantar una Gestión de Riesgos adecuada será un elemento decisivo a la hora de asegurar el Proyecto, mediante la identificación y el análisis por adelantado de los riesgos potenciales que puedan afectar al Proyecto, y la elaboración de las acciones de contingencia adecuadas para evitar su aparición o para minimizar el impacto en el Proyecto, en caso de que finalmente el riesgo se verifique.

### 1.1. Propósito

Este documento presenta el análisis de los riesgos identificados durante la fase de Inicio del proyecto “Implementación de un Sistema de Control de Sacramentos y Asignación de Personal”. Para cada riesgo observado se valorarán sus efectos y contexto de aparición para el caso en que se convierta en un hecho. Además, se definirán estrategias para reducir la probabilidad del riesgo o para controlar sus posibles efectos.

### 1.2. Alcance

El ámbito del análisis de riesgos cubre toda la extensión del proyecto observado desde su fase inicial. Será necesario durante el desarrollo del proyecto revisar y actualizar los contenidos del análisis de riesgos en caso de que se detecten nuevos riesgos no visibles en este momento.

Este documento será aplicable a todas las fases del Proyecto.

## 2. Gestión del Riesgo

### 2.1. Identificación de Riesgos

Listado de Riesgos, Tipo de Riesgo

	Descripción del Riesgo	Tipo de Riesgo
R01	Requisitos poco claros	Riesgo del Producto
R02	Falta de Experiencia en tareas de planificación	Riesgo del Proyecto
R03	Falta de Experiencia con las herramientas utilizadas	Riesgo del Producto/Proyecto
R04	Diseño Erróneo	Riesgo del Producto
R05	Falta de un Experto	Riesgo del Proyecto
R06	Pérdida de documentación y/o otros artefactos	Riesgo del Proyecto
R07	Control de Calidad inadecuado	Riesgo del Proyecto
R08	Estimación de costos fuera del alcance de la realidad	Riesgo del Proyecto
R09	Falta de seguimiento permanente de tareas y actividades	Riesgo del Proyecto
R10	Excesivo tiempo en corregir errores	Riesgo del Producto

### 2.2. Análisis del Riesgo

ID	Análisis del Riesgo
R01	<p><b>Magnitud</b></p> <p>Variable según la fase de aparición:</p> <ul style="list-style-type: none"><li>▪ Inicio: baja.</li><li>▪ Elaboración: media.</li></ul>

	<ul style="list-style-type: none"> <li>▪ Construcción: alta.</li> <li>▪ Transición: muy alta</li> </ul> <p><b>Descripción</b> Los requisitos representan la idea que tiene el cliente sobre la aplicación, sobre ellos se construyen los casos de uso y dichos casos de uso guían el desarrollo del proyecto. Una mala o insuficiente recolección de los mismos afecta a la calidad de todo el proyecto.</p> <p><b>Impacto</b> La incorporación o modificación de requisitos durante el desarrollo requerirá realizar cambios sobre gran parte de la documentación del producto elaborada con anterioridad al momento del cambio. Estas modificaciones serán menos costosas durante las dos primeras fases del proyecto, pero pueden suponer trastornos importantes durante las fases de Construcción y Transición, pues no sólo cambiaría la documentación sino también el código fuente y los ejecutables.</p> <p><b>Indicadores</b> Al realizar la consulta al cliente, no sabe indicar con propiedad cuales son los servicios que espera obtener de la aplicación.</p>
R02	<p><b>Magnitud</b> Media.</p> <p><b>Descripción</b> El responsable tiene poca experiencia en el desarrollo de software siguiendo una estructura de tareas y fechas preestablecido.</p> <p><b>Impacto</b> La planificación guía todo el desarrollo del proyecto. Un error en la misma puede incidir directamente en sus resultados. No obstante, la división en iteraciones reduce el posible impacto de los errores, permitiendo que estos puedan ser corregidos o absorbidos en iteraciones posteriores a la de su aparición.</p> <p><b>Indicadores</b> Diferencias entre el desarrollo real del proyecto y la planificación estimada.</p>
R03	<p><b>Magnitud</b> Variable según la fase de aparición:</p> <ul style="list-style-type: none"> <li>▪ Inicio: baja.</li> <li>▪ Elaboración: media.</li> <li>▪ Construcción: alta.</li> <li>▪ Transición: alta.</li> </ul> <p><b>Descripción</b> El responsable tiene dificultades a la hora de realizar sus objetivos (tanto de documentación como de implementación) por su inexperiencia con las herramientas disponibles para el mismo.</p> <p><b>Impacto</b> Puede suponer retrasos.</p> <p><b>Indicadores</b> No procede.</p>

R04	<p><b>Magnitud</b> Baja en Elaboración, alta en Construcción.</p> <p><b>Descripción</b> El diseño del sistema resulta inadecuado. Al realizar actividades de implementación puede encontrarse que el diseño carece del suficiente nivel de detalle o está mal enfocado, bien por la naturaleza del problema, o bien por restricciones de uso impuestas por tecnologías de terceros.</p> <p><b>Impacto</b> Puede introducir retrasos en el proyecto ante la necesidad de volver a considerar el diseño trazado. Requiere la actualización o modificación de los artefactos de diseño.</p> <p><b>Indicadores</b> La arquitectura no cumple las expectativas. Se complica la implementación.</p>
R05	<p><b>Magnitud</b> Media.</p> <p><b>Descripción</b> No hay un experto del dominio en el equipo de consultoría o en el desarrollo, al que poder consultar.</p> <p><b>Impacto</b> Puede suponer retrasos.</p> <p><b>Indicadores</b> No procede</p>
R06	<p><b>Magnitud</b> Alta.</p> <p><b>Descripción</b> Por causas varias se pierde parte o el total de la documentación así como la posibilidad de perder parte o el total de otros artefactos, como pueden ser: parte de la implementación o ficheros de planificación.</p> <p><b>Impacto</b> Variable, puede suponer una catástrofe, o un simple retraso.</p> <p><b>Indicadores</b> Ninguno.</p>
R07	<p><b>Magnitud</b> Alta.</p> <p><b>Descripción</b> Control de calidad escaso o inapropiado, por falta de experiencia y/o conocimientos.</p> <p><b>Impacto</b> Si no se lleva un control de calidad correcto durante la construcción del producto, esto puede ocasionar que la calidad de este sea diferente al que se planeó, lo cual puede ocasionar que el cliente rechace el producto y este tenga que ser</p>

	<p>rediseñado, utilizando tiempo que le corresponde a otras actividades.</p> <p><b>Indicadores</b> Durante el desarrollo del producto no se le da importancia al control de calidad.</p>
R08	<p><b>Magnitud</b> Media</p> <p><b>Descripción</b> Se sobreestiman o subestiman los costos involucrados con el desarrollo del producto de software</p> <p><b>Impacto</b> Puede generar que el equipo entre en períodos de sobrecarga de trabajo o periodos de ausencia del mismo, lo que a su vez puede conllevar a un deterioro en la calidad</p> <p><b>Indicadores</b> El equipo trabaja más o menos horas de las inicialmente programadas, se presentan quejas a jefe del Proyecto o Pedidos de redimensionamiento</p>
R09	<p><b>Magnitud</b> Media</p> <p><b>Descripción</b> No se realiza un seguimiento de las tareas planificadas, lo que puede ocasionar que algunas de ellas sean dejadas para última instancia, con la consecuente baja en su calidad</p> <p><b>Impacto</b> Sobrecarga de trabajo en los días previos a la entrega de un entregable, pobre calidad de los entregables, se obvian detalles importantes.</p> <p><b>Indicadores</b> No se revisa la documentación de planificación y si se revisa no se le da la debida importancia al seguimiento de actividades planeadas ni al resultado.</p>
R10	<p><b>Magnitud</b> Alta</p> <p><b>Descripción</b> La corrección de los errores encontrados en las pruebas lleva más tiempo del planeado.</p> <p><b>Impacto</b> Puede generar retrasos así como también que se vuelvan a desarrollar módulos que ya se encontraban terminados.</p> <p><b>Indicadores</b> El cliente y/o el jefe de proyecto anuncia al equipo el cambio de tecnología.</p>

### 2.3. Acciones de Prevención y de Corrección

ID	Plan de Prevención	Plan de Corrección
R01	Realización de reuniones con el cliente; elaboración de cuestionarios para aclarar puntos poco claros de las reuniones previas.	<p>En las primeras fases se realizarán los cambios necesarios para incorporar los nuevos requisitos o los cambios necesarios para que se cumpla con la funcionalidad solicitada. En las fases de Construcción y Transición se valorará la importancia de las modificaciones/requisitos nuevos frente a la cantidad de tiempo disponible para abordarlos.</p> <p>En caso de que se decida aceptarlos, se revisarán los requisitos afectados, así como toda la documentación y código derivado de los mismos hasta el punto de aparición del cambio.</p>
R02	El uso de RUP como proceso de desarrollo. Realización de reuniones entre los miembros del proyecto para la evaluación de la marcha del proyecto y consultas al tutor.	Se observarán las diferencias entre la planificación de cada iteración y el informe de seguimiento de su ejecución, analizando las causas de sus diferencias para tratar de detectar y corregir errores de planificación en las iteraciones posteriores.
R03	Una parte del tiempo de desarrollo del proyecto se destinará al aprendizaje de las nuevas herramientas.	Si se produce un retraso en el aprendizaje por parte de un miembro del equipo, los demás miembros tratarán de ayudar a superarlo. Si no resultara, consultar a fuentes externas como profesores, bibliografía, foros en Internet. En último lugar se haría una redistribución de tareas.
R04	Durante la fase de Elaboración se desarrollará en paralelo un prototipo contenido la arquitectura del sistema para comprobar la validez de la misma. En caso de encontrarse errores o inconsistencias, podrá modificarse el diseño al mismo tiempo que la	<p>Si el riesgo se convierte en hecho durante la fase de Elaboración, se revisará y modificará la documentación de diseño afectada.</p> <p>Si lo hace durante la fase de construcción, se estudiará una solución acorde a los tiempos de plazo de que se dispone.</p> <p>La planificación se reajustará si fuera necesario.</p>

	implementación del prototipo.	
R05	Aprendizaje continuo durante todo el proyecto	Las dudas que no se sepan resolver se trasladarán al tutor y a foros especializados.
R06	Se realizarán copias de seguridad en los dispositivos externos a disposición, así como también el alojamiento en la nube.	Actualizar con la última copia disponible
R07	Comprobar si se están siguiendo los estándares fijados en la planificación así como también los requerimientos establecidos.	Se estudiará posibles soluciones teniendo en cuenta el tiempo que se tiene para el proyecto. Si es necesario se harán reajustes a los documentos involucrados.
R08	Realizar estimaciones en base a varias herramientas para tratar de hallar un estimado más cercano a la realidad	Redimensionar el proyecto conforme se va desarrollando y nuevas funcionalidades se agregan o se eliminan.
R09	Llevar al día una revisión del estado del proyecto para anotar los posibles atrasos y poder así tomar medidas en el instante.	Realizar una recandelerización de tareas, así como llamadas de atención a los miembros del equipo que dejen sus tareas para última instancia.
R10	Uso de un estándar de programación y/u ordenamiento del mismo para ubicar fácilmente trozos de código que ejecuten cada acción.	Se acudirá a copias de seguridad antes de codificar la parte que genera errores, de este modo si no puede ubicarse se puede codificar de nuevo teniendo como modelo el código con error e identificar estos errores fácilmente. División del código y prueba de cada trozo por separado para identificar en que parte está el error.

## 2.4. Control y Seguimiento de Riesgos

Id.	Responsable	Fecha de Terminación	Estado	Observaciones
-----	-------------	----------------------	--------	---------------

R01	Jefe de Proyecto	Fin del Proyecto	Iniciado	
R02	Jefe de Proyecto	Fin del Proyecto	Iniciado	
R03	Equipo de Desarrollo	Fin del Proyecto	Iniciado	
R04	Jefe de Proyecto	Fin del Proyecto	Iniciado	
R05	Equipo de Desarrollo	Fin del Proyecto	Iniciado	
R06	Equipo de Desarrollo	Fin del Proyecto	Iniciado	
R07	Jefe de Proyecto	Fin del Proyecto	Iniciado	
R08	Jefe de Proyecto	Fin del proyecto	Iniciado	
R09	Jefe del proyecto	Fin del proyecto	Iniciado	
R10	Equipo de Desarrollo	Fin del Proyecto	Iniciado	

**Responsable:** Persona o personas asignadas a la implantación de las acciones preventivas y/o correctoras

**Fecha Terminación:** Fecha límite en la cual todas las acciones anteriormente descritas deban haber sido ejecutadas por el responsable o responsables asignados.

**Estado:** Estado Actual del Riesgo y de las Acciones Preventivas y/o Correctoras.

**Observaciones:** Descripción de las observaciones encontradas para este riesgo (opcional).

### 3. Matriz de Riesgo

Se propone la utilización de una matriz específica que sirva de soporte para la Gestión de Riesgos. Esta matriz se utilizará en las reuniones de seguimiento y/o cuando se estime necesario (en el caso de situaciones excepcionales), y su contenido será el siguiente:

<b>Id.</b>	<b>Descripción del Riesgo</b>	<b>Tipo Riesgo</b>	<b>Probab. Ocurrencia</b>	<b>Nivel de Impacto</b>	<b>Evaluación del Riesgo</b>	<b>Acciones de Prevención</b>	<b>Acción de Corrección</b>
R01	Cambios en los Requisitos	Producto	20	4	0.8	Realización de reuniones con el cliente para la aclaración de requisitos.	Se revisarán los requisitos afectados, así como toda la documentación y código derivado de los mismos hasta el punto de aparición del cambio.
R02	Falta de Experiencia en tareas de planificación	Proyecto	50	2	1	El uso de RUP como proceso de desarrollo. Realización de reuniones entre los miembros del proyecto para la evaluación de la marcha del proyecto y consultas al tutor.	Se observarán las diferencias entre la planificación de cada iteración y el informe de seguimiento de su ejecución, para tratar de detectar y corregir errores de planificación en las iteraciones posteriores.
R03	Falta de Experiencia con las herramientas utilizadas	Producto/ Proyecto	50	2	1	Una parte del tiempo de desarrollo del proyecto se destinará al aprendizaje de las herramientas de documentación e implementación.	Si se produce un retraso por parte de un miembro del equipo, los demás miembros tratarán de ayudar a superarlo. Consultar a fuentes externas. En último lugar se haría una redistribución de tareas.
R04	Diseño Erróneo	Producto	40	3	1.2	Durante la fase de Elaboración se desarrollará en paralelo un prototipo conteniendo la arquitectura del sistema para comprobar la validez de la misma.	Se revisará y modificará la documentación de diseño afectada. La planificación se reajustará si fuera necesario.
R05	Falta de un Experto	Proyecto	80	1	0.8	Aprendizaje continuo durante todo el proyecto	Las dudas que no se sepan resolver se trasladarán al tutor y a foros especializados.
R06	Pérdida de documentaci	Proyecto	40	4	1.6	Se realizarán copias de seguridad en los dispositivos externos a disposición, así como también el	Actualizar con la última copia disponible

	ón y/o otros artefactos					alojamiento en la nube.	
R07	Control de Calidad inadecuado	Proyecto	60	4	2.4	Comprobar si se están siguiendo los estándares fijados en la planificación así como también los requerimientos establecidos.	Se estudiará posibles soluciones teniendo en cuenta el tiempo que se tiene para el proyecto. Si es necesario se harán reajustes a los documentos involucrados.
R08	Mala estimación de costos	Proyecto	50	3	1.5	Realización de varias estimaciones con metodologías diferentes.	Redimensionar el proyecto conforme se ejecuta.
R09	Falta de seguimiento de tareas	Proyecto	50	3	1.5	Llevar al día una revisión del estado del proyecto para anotar los posibles atrasos y poder así tomar medidas en el instante.	Recandelerización de las tareas, charla con el equipo de desarrollo en caso de detectarse malas prácticas.
R10	Excesivo tiempo en corregir errores	Producto	40	3	1.2	Uso de un estándar de programación y/u ordenamiento del mismo para ubicar fácilmente trozos de código que ejecuten cada acción.	Se acudirá a copias de seguridad antes de codificar la parte que genera errores, de este modo si no puede ubicarse se puede codificar de nuevo teniendo como modelo el código con error e identificar estos errores fácilmente. División del código y prueba de cada trozo por separado para identificar en que parte está el error.

**Id.:** Identificador de Riesgo

**Descripción del Riesgo:** Descripción Resumida del Riesgo

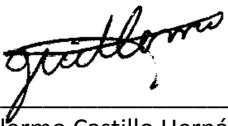
**Probabilidad** (1 a 100): Grado de probabilidad de que el Riesgo finalmente se produzca. Se mide en una escala de 1 a 100 (porcentual).

**Nivel de Impacto:** Grado de Impacto en el Proyecto en el caso de que el Riesgo finalmente se produjera. Se mide en una escala de 1 a 5, siendo 1=poco influyente hasta 5=fuertemente influyente.

**Probabilidad Ocurrencia:** Valor numérico resultante del producto del Grado de Probabilidad por el Grado de Impacto. Este producto dará la prioridad que tendrá la gestión de este Riesgo y la implantación de sus medidas preventivas o correctoras.

**Acciones Prevención:** Descripción de las Acciones o Medidas a Adoptar para evitar (mitigar) la aparición final del Riesgo

**Acciones Corrección:** Descripción de las Acciones o Medidas a Adoptar en el caso en el que el Riesgo finalmente se haya producido

APROBACIONES	
 Juan Guillermo Castillo Hernández Gerente del Proyecto 06/07/2013	 Vicente Diez Varona Párroco Patrocinador del Py 06/07/2013

Anexo Nº 3



**OBISPADO DE ICA**

Nº 004740

**CONSTANCIA DE BAUTISMO**

Por la presente se certifica que en el Archivo de la

Parroquia (Capellánía, etc.) \_\_\_\_\_

Se encuentra una Partida en el Libro de Bautismo N° 118, a fojas 239

Número 473 cuyos datos son:

Apellidos \_\_\_\_\_

Nombres LISBETH YERALDIN.-

Padre \_\_\_\_\_

Madre Rocio Rosario Pardo Palomino.-

Lugar y Fecha de Nacimiento Pisco, El dia 11 de Mayo de 1,988.-

Fecha de Bautismo El dia 10 de Noviembre de 1,990.-

Padrino Walter Hurtado Alarco.-

Madrina Alicia de Hurtado.-

Ministro P. Alfonso García C.M.-

Nota Marginal NINGUNA

PISCO \_\_\_\_\_ de \_\_\_\_\_ de 20\_\_\_\_\_

PARROCO  
P. José E. Torre Motta C.M.

MISSIONARIO ARGENTINO



El suscrito certifica la autenticidad de la firma y sello de la presente Constancia de Bautismo.

de \_\_\_\_\_ de 20\_\_\_\_\_

Notario Eclesiástico

Sello de la Curia

Anexo Nº 4



**OBISPAZO DE ICA**

**CONSTANCIA DE CONFIRMACIÓN**

001190

Por la presente se certifica que en el Archivo de la  
Parroquia (Capellanía, etc.) SAN CLEMENTE DE PISCO

Se encuentra una Constancia de Confirmación cuyos datos son:

Apellidos CUSIPUMA MUÑANTE.  
Nombres PATRICIA ROSITA.  
Padre Luis Cusipuma.  
Madre Rosa Muñante.  
Parroquia de Bautismo San Francisco de Asís.  
Fecha de Confirmación El dia 07 de Diciembre del 2.003.-  
Padrino o Madrina Flavia Guillen Vera.  
Ministro Mons. Guido Breña López O.P.  
Anotaciones Ninguna

PISCO 13 de MARZO de 20 12

  
(Sello)

Él suscrito certifica la autenticidad de la firma y sello de la presente Constancia de Confirmación.

Notario Ecclesiástico

Sello de la Curia

Anexo Nº 5



**OBISPADO DE ICA**

**CONSTANCIA DE MATRIMONIO**

001414

Por la presente se certifica que en el Archivo de la  
**SAN CLEMENTE DE PISCO**

Parroquia (Capellania, etc.)

Se encuentra una Partida en el Libro de Matrimonios N° 5, a fojas 76

Número 143 cuyos datos son:

Esposo PEDRO YIKA CARASACULLO.-

Nacido en GRECIA.-

Esposa FLAVIA FALCONI DE LA CRUZ.-

Nacida en PISCO.-

Fecha de Matrimonio El dia 01 de Mayo de 1,885.-

Padrinos \_\_\_\_\_

Testigos Fermin Tioiciano e Ysidoro Jaimes.-

Sacerdote Asistente P. Pedro Falcon Escobar.-

Anotaciones Ninguna

PISCO, 28 de Mayo de 20 10

(Sello)

El suscrito certifica la autenticidad de la firma y sello de la presente Constancia de Matrimonio.

\_\_\_\_\_ de \_\_\_\_\_ de 20 \_\_\_\_

Notario Ecclesiastico

Órgano Judicial