

**UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE HUAMANGA**

**FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**“APLICACIÓN WEB Y MÓVIL PARA FONDO DOCUMENTAL DE LA  
ARQUIDIÓCESIS DE AYACUCHO, 2019”**

**Tesis presentada por** : Bach. Fiorella Luque Mendieta

**Tipo de investigación** : Observacional, retrospectivo, descriptivo

**Área de investigación** : Ingeniería de Software

**Asesor** : MSc. Ing. Efraín Elías Porras Flores

Ayacucho – Julio

2019

## **AGRADECIMIENTO**

A la universidad Nacional de San Cristóbal de Huamanga, mi alma mater, a todos los docentes que de alguna forma han contribuido en mi formación profesional.

## **DEDICATORIA**

A mi madre Rosario y abuela María por ayudarme y apoyarme siempre con sus consejos y su ejemplo de perseverancia, rectitud, integridad y ética.

A mi hijo Felipe por ser mi orgullo y motivación en mí, día a día.

A mis hermanos por estar conmigo y por la paciencia que me han tenido.

A mis amigos que compartieron los años de aprendizaje, y que me apoyaron para concluir mis metas.

## CONTENIDO

AGRADECIMIENTO.....	i
DEDICATORIA.....	ii
CONTENIDO.....	iii
RESUMEN.....	v
INTRODUCCIÓN.....	vi

### CAPÍTULO I

#### PLANTEAMIENTO DE INVESTIGACIÓN

1.1. DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA.....	1
1.2. PROBLEMA DE INVESTIGACIÓN.....	1
1.3. OBJETIVOS DE LA INVESTIGACIÓN.....	2
1.4. JUSTIFICACIÓN Y DELIMITACIÓN DE LA INVESTIGACIÓN.....	2
1.4.1. JUSTIFICACIÓN.....	2
1.4.2. DELIMITACIÓN.....	4

### CAPÍTULO II

#### REVISIÓN LITERARIA

2.1. ANTECEDENTES.....	5
2.2. MARCO TEÓRICO.....	6
2.2.1. ARCHIVO HISTORICO.....	6
2.2.2. FONDO DOCUMENTAL.....	6
2.2.3. PROGRAMACION EXTREMA.....	10
2.2.4. APLICACIÓN WEB.....	23
2.2.5. APLICACIÓN MÓVIL.....	23
2.2.6. PROGRAMACIÓN ORIENTADA A OBJETOS.....	24
2.2.7. TECNOLOGÍAS DE INTERNET.....	25
2.2.8. SISTEMA DE GESTIÓN DE BASE DE DATOS.....	27
2.2.9. POBLACIÓN Y MUESTRA.....	28
2.2.10 MUESTREO.....	29

### CAPÍTULO III

#### METODOLOGÍA DE LA INVESTIGACIÓN

3.1. TIPO DE INVESTIGACIÓN.....	30
3.2. NIVEL DE INVESTIGACIÓN.....	30
3.3. DISEÑO DE LA INVESTIGACIÓN.....	30
3.4. POBLACIÓN Y MUESTRA.....	30

3.5.	VARIABLES E INDICADORES.....	31
3.6.	TÉCNICAS E INSTRUMENTOS PARA RECOLECTAR DATOS.....	32
3.7.	HERRAMIENTAS PARA EL TRAMIENTO DE DATOS.....	32
3.8.	TÉCNICAS PARA APLICAR PROGRAMACIÓN EXTREMA.....	34

#### **CAPÍTULO IV**

##### **ANÁLISIS Y RESULTADOS DE LA INVESTIGACIÓN**

4.1.	ARTEFACTOS DEL SOFTWARE APLICANDO PROGRAMACIÓN EXTREMA .....	37
4.1.1.	FASE DE EXPLORACIÓN.....	37
4.1.2.	FASE DE PLANIFICACIÓN.....	40
4.1.3.	FASE DE ITERACIÓN.....	45

#### **CAPÍTULO V**

##### **CONCLUSIONES Y RECOMENDACIONES**

5.1.	CONCLUSIONES.....	72
5.2.	RECOMENDACIONES.....	72
	BIBLIOGRAFÍA.....	73
	ANEXOS.....	77

## **RESUMEN**

El Archivo Arzobispal de Ayacucho, custodia importante documentación histórica desde su creación en 1609. Estuvo funcionando y ofreciendo sus servicios al público hasta la década del 90, pero hubo hurtos y deterioro de distintos documentos de la historia de Huamanga y del Perú, la búsqueda de los documentos históricos se hace difícil porque son documentos que en la mayoría de los casos están deteriorados, la búsqueda manual seguirá deteriorando aún más estos archivos, perdiendo la historia de Ayacucho.

El desarrollo de un software para administrar el fondo documental de la arquidiócesis en el arzobispado de Ayacucho, permitirá conservar los documentos físicos y el acceso en tiempo real a la información histórica; realizando un estudio de tipo observacional, retrospectivo y descriptivo, de nivel descriptivo, de los archivos eclesiásticos, aplicando la técnica de análisis documental para cada documento y la identificación de las necesidades de búsqueda de los usuarios externos e internos, brindando la documentación eclesial al servicio de la comunidad.

Se ha desarrollado un software web y móvil para administrar el fondo documental, aplicando el proceso ágil programación extrema, el lenguaje de programación java, un administrador de base de datos relacional, para acceder en tiempo real a la información del fondo documental de la Arquidiócesis de Ayacucho. La investigación se realizó en el Área documental de la Arquidiócesis donde se almacena los archivos históricos.

Se ha logrado automatizar el fondo documental bautismo, mediante las historias de usuario 1, 6 y 10, interfaz gráfica figura 4.8 y 4.9; matrimonio, con las historias de usuario 2, 7 y 10 interfaz gráfica figura 4.8 y 10; defunciones, con las historias de usuario 3, 8 y 10, interfaz gráfica figura 4.8 y 4.11; confirmación, con las historias de usuario 4, 9 y 10, interfaz gráfica figura 4.8 y 4.12 la base de datos relacional y el código fuente correspondiente, que permite registrar, modificar y consultar los bautizos, matrimonio, defunciones y confirmación.

Palabras clave: Fondo documental, Archivo histórico, Programación extrema.

## INTRODUCCIÓN

Según González (2010), el usuario se enfrenta al problema de encontrar mecanismos, o instrumentos de localización del fondo documental, que sobre el papel le permitan detectar la información idónea para adelantar su trabajo. Normalmente, estos instrumentos de descripción, guías, catálogos o inventarios se encuentran en las salas o servicios de investigación. De tal modo que, frecuentemente, un investigador que había proyectado una estancia corta en un archivo, con los problemas logísticos que ello conlleva, se enfrenta con la necesidad de asumir que una parte importante de su estancia, al contrario de lo que tenía planificado, debe dedicarla exclusivamente a localizar los documentos que necesita, o a preparar con mayor precisión los criterios y términos de búsqueda de los textos que, en teoría, le ofrecerán la información histórica original necesaria.

Los fondos documentales se define como “conjunto de documentos, de cualquier formato o soporte, producidos orgánicamente y/o reunidos y utilizados por una persona particular, familia u organismo en el ejercicio de sus actividades" (Archivo Nacional de Chile, 2016).

La programación extrema (XP) es una metodología de desarrollo ligera (o ágil) basada en una serie de valores y de prácticas de buenas maneras que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas. (Aguilar, 2012).

El presente estudio es para implementar un software web y móvil para administrar el fondo documental de la arquidiócesis de Ayacucho, actualmente no existe software que permita obtener información de los archivos documentales en línea y acceder en tiempo real a toda la información histórica.

Los objetivos específicos son: Automatizar la información histórica de los documentos físicos sobre bautizos, matrimonios, defunciones y confirmación, con la finalidad de acceder a los documentos digitales en tiempo real y, que sirva como base para trámites administrativos y legales.

# **CAPÍTULO I**

## **PLANTEAMIENTO DE LA INVESTIGACIÓN**

### **1.1. DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA**

El Archivo Arzobispal de Ayacucho alberga documentos que datan del siglo XVI hasta fines del siglo XIX, entre ellos partidas de bautismo, matrimonio, defunción y pliegos matrimoniales; así como también documentación histórica generada por el Arzobispado desde el año 1609.

No existe la planificación descriptiva de los archivos eclesiásticos en la Arquidiócesis de Ayacucho, que debe incluir las características de cada documento y la identificación de las necesidades de búsqueda de los usuarios externos e internos. Al ser este un archivo que contiene rica información, necesitan un tratamiento archivístico diferente al de cualquier otro archivo que se complemente con el valor histórico, informativo y probatorio que tiene el archivo, que pertenece al patrimonio cultural de Ayacucho.

Toda la documentación generada por la Iglesia Ayacuchana en el ejercicio de sus funciones es el signo y testimonio de la rica vida eclesial, la Iglesia en Ayacucho debe cuidar la rica documentación que genera todas las fases del ciclo vital de los documentos y en todos los soportes. Enfrentar el tratamiento de la documentación eclesial con las técnicas adecuadas y ponerlo a disposición del público, que permite cumplir con uno de los fines de una institución cuyos planes de evangelización, entre otros objetivos, buscan concretar la relación entre fe y cultura. Una forma inmejorable de combinar esos dos conceptos es, poner la documentación eclesial al servicio de la comunidad en la que se inserta con las demandas requeridas por la evolución de la sociedad de la información.

### **1.2. PROBLEMA DE INVESTIGACIÓN**

#### **PROBLEMA PRINCIPAL**

¿Cómo obtener información del fondo documental de la Arquidiócesis de Ayacucho, 2019?.

#### **PROBLEMAS SECUNDARIOS**



¿De qué manera acceder a la información histórica de los; bautizos, matrimonios, defunciones y confirmación?.

### **1.3. OBJETIVOS DE INVESTIGACIÓN**

#### **OBJETIVO GENERAL**

Desarrollar una aplicación web y móvil para publicar el fondo documental, mediante técnicas e instrumentos; aplicando el proceso ágil programación extrema, un lenguaje de programación orientado a objetos, un administrador de base de datos relacional, con la finalidad de acceder en tiempo real a la información histórica de la Arquidiócesis de Ayacucho, 2019.

#### **OBJETIVOS ESPECÍFICOS**

Automatizar la información histórica de los documentos físicos sobre bautizos, matrimonios, defunciones y confirmación, con la finalidad de acceder a los documentos digitales en tiempo real y, que sirva como base para trámites administrativos y legales.

### **1.4. JUSTIFICACIÓN Y DELIMITACIÓN DE LA INVESTIGACIÓN**

#### **1.4.1. JUSTIFICACIÓN**

#### **TÉCNICA**

Se plantea desarrollar un software para la publicación digital de los archivos históricos del arzobispado de Ayacucho, para brindar un índice temático, con el cual podremos acceder de manera fácil y en tiempo real a los fondos archivísticos en series documentales y catálogos, digitalizados. Como resultado de este software, se dará el acceso a los investigadores y usuarios a las fuentes documentales, permitiéndoles su revisión y estudio de forma rápida y oportuna, mediante el despacho de imágenes digitales e información que este software brindará.

El software puede ser usado en otro archivo histórico de los diferentes arzobispados y diócesis del Perú. El uso de las tecnologías de la información y comunicaciones permitirá que los investigadores localicen la información de los archivos. Es evidente que el uso de las tecnologías proporciona mayor comodidad, control y seguridad, ahí donde el investigador puede consultar directamente por Internet, mediante sistemas de consulta de

bases de datos, los fondos documentales del archivo, creando una relación entre el usuario o investigador, la calidad de la interfaz de consulta, el volumen de información y la rapidez de acceso.

El software no solo puede ser utilizado en entidades religiosas y otros. Sino, también sirve de guía de desarrollo para docentes y estudiantes universitarios, de la especialidad de ingeniería de sistemas. Quienes deben ser entendidos como seres dinámicos, responsables y proactivos de su proceso de aprendizaje, que utilizan los contenidos de la investigación para incorporarlos, no solo a su vida personal, sino también, a su contexto académico.

## **SOCIAL**

El software, de cara a facilitar el acceso de los investigadores a las fuentes documentales y con ellos a toda la sociedad pueden venir dadas por la optimización en la oferta de los instrumentos descriptivos clásicos de los archivos históricos, así como, por la implantación de un paquete de datos adecuado a las necesidades propias de la gestión del archivo, y a garantizar la consulta fácil de los documentos del archivo arzobispal. Además pretende mejorar el manejo integral de la gestión del archivo histórico arzobispal, mediante un servicio de digitalización de los diferentes fondos documentales conservando de esta manera los documentos originales, de robos y deterioros protegiendo así nuestro patrimonio cultural.

La importancia del software sirve para conservar los archivos que son imprescindibles para los historiadores de los hechos religiosos. En definitiva, sea por la investigación que fuere, estos fondos documentales contienen una riqueza inigualable y por la dificultad que entraña su acceso, es importante digitalizarlos para evitar su deterioro y obtenerlos en formatos digitales.

## **ECONÓMICA**

La propuesta de este software es apoyar a la gestión del archivo histórico del Arzobispado de Ayacucho, haciendo una buena presentación que les permitan desarrollar un plan de control a los usuarios evitando costos de instalaciones nuevas, robos y deterioro de los fondos documentales.

Este software apoyará al Arzobispado, ya que permitirá brindar los servicios del archivo histórico al público en general, permitiéndole un ingreso económico por el uso y

solicitudes de copias de documentación relevante al usuario. Los problemas de la conservación de los documentos históricos son una de las grandes limitantes en la gestión archivística, ya que requieren inversiones cuantiosas debido a la restauración, adecuado manejo y control de calidad de fondos documentales. En el software se muestra como desde el Área Documental del Arquidiócesis de Ayacucho se implementara nuevas herramientas informáticas para la unidad del archivo histórico. Éstas nos permiten asegurar el acceso a la totalidad de los fondos que conservan de una forma rápida y rigurosa para todos los ciudadanos que acceden a los servicios archivísticos. Al mismo tiempo esta apuesta, en un contexto de fuerte crisis económica, asegura la reducción en los costos de mantenimiento informático sin renunciar a la calidad en la gestión archivística.

#### **1.4.2. DELIMITACIÓN**

El estudio se realiza sobre las series documentales: bautizos, matrimonios, defunciones y confirmación de los fondos documentales de la arquidiócesis de Ayacucho, 2019.

## **CAPITULO II**

### **REVISIÓN DE LITERATURA**

#### **2.1. ANTECEDENTES**

Según la investigación desarrollada por Tellez (2014) se concluye que; En el centro excursionista de Cataluña se ha puesto en práctica nuevas herramientas informáticas en software libre para sistematizar los fondos documentales que contienen su institución, de tal manera que la población pueda acceder a la totalidad de los fondos documentales de manera rápida y en menos tiempo y al paralelo genera menos gastos económicos en el mantenimiento informático sin afectar la buena gestión del fondo documental.

La digitalización de los documentos sacramentales a partir del sistema de archivo, gestión y difusión de todas y cada una de las partidas de Bautismo, Matrimonio y Defunción, permite al usuario acceder de manera rápida y en tiempo real los extractos importantes del contenido del documento que requiere. Esta mutua colaboración entre la administración y la iglesia puede ser utilizada para otras diócesis, ya que los valiosos fondos documentales religiosos pueden ser puestos en disposición para los creyentes o no (Garro, 2007).

La nueva Guía de los Archivos de la Iglesia en España, es una colección de “Instrumentos Informáticos de Consulta de los Archivos de la Iglesia en España”, describe de manera detallada los ciento setenta fondos documentales eclesiásticos de España como los catedralicios, los diocesanos, los monásticos y los conventuales, también resume los registros de los libros de bautizos, confirmación, matrimonios, defunciones y otros fondos documentales.

La asociación de archiveros de la iglesia española, hace una publicación en el año 1985 de una primera guía de los archivos de la iglesia en España, donde se realiza una descripción detallada de los archivos mayores y las bibliotecas eclesiásticas, esta guía mediante revisiones se ha actualizado y presentado en formato CD para su nueva y final revisión para que posteriormente sea interactiva a través del internet.

Posteriormente se ha actualizado la guía de los Archivos de la Iglesia en España en (Vol. II), con el título de Memoria Ecclesiae, donde se accede de manera inmediata a los fondos documentales de la iglesia de España. Desarrollando las nuevas tecnologías informáticas se piensa aumentar en gran medida la divulgación de la información de las Series documentales de los archivos eclesiásticos de España y ser más eficiente en favor de la investigación. (Instrumentos informáticos de consulta de los archivos de la Iglesia en España, 2001).

## **2.2. MARCO TEÓRICO**

### **2.2.1. ARCHIVO HISTÓRICO**

Es donde se encuentra la documentación de carácter permanente, más de 30 años de antigüedad, que sirven para la investigación científica e histórica (Cortés, 1982).

Archivo histórico, es el archivo al cual se transfiere del archivo central o del archivo de gestión, la documentación que por decisión del correspondiente Comité de Archivo, debe conservarse permanentemente, dado el valor que adquiere para la investigación, la ciencia y la cultura. Este tipo de archivo también puede conservar documentos históricos recibidos por donación, depósito voluntario, adquisición o expropiación. (Romero, 1994).

### **2.2.2. FONDO DOCUMENTAL**

Se entiende por fondo documental la agrupación orgánica de documentos generados por una institución o persona física o jurídica en el ejercicio de sus funciones que constituyen la expresión del conjunto de las actividades desarrolladas (Real Academia de la Lengua Española, 2001).

Los fondos documentales constituyen la mayor agrupación documental existente en un archivo, y corresponden al "conjunto de documentos, de cualquier formato o soporte, producidos orgánicamente y/o reunidos y utilizados por una persona particular, familia u organismo en el ejercicio de sus actividades" (Archivo Nacional de Chile, 2016).

Fondo es un conjunto de documentos producidos en un contexto relacional normalizado, y su función es informar y probar; la colección es la reunión de documentos como fruto del saber y de la voluntad creadora, y su función es informar. El primero se entiende agrupado en la secuencia de su producción, mientras que la segunda puede ser el

resultado de la alteración o destrucción de la relación natural y justa entre el todo y la parte, que motiva la transformación del fondo archivístico en una colección de piezas “importantes y valiosas” que sirven para unos fines determinados, pero no a la memoria total de una sociedad o de un individuo en sus aspectos más diversos (Gómez, s.f.).

## **A. BAUTISMO**

El bautismo es un sacramento por el que se acoge al neófito en la fe cristiana al canon 877.1, que establece “El párroco del lugar en el que se celebre el bautismo debe anotar diligentemente y sin demora en el libro de bautismo el nombre del bautizado, haciendo mención del ministro, los padres, padrinos, testigos si los hubo y el lugar y día en que se administró, indicando asimismo el día y lugar del nacimiento”. Estos datos que hoy se incluyen en los asientos bautismales, poco tienen que ver con los que se hicieron en su origen y que se reducían al nombre del párroco, fecha del bautizo, y nombre del bautizado y sus padres. Hasta principios del siglo XIX, también era habitual incluir en ellos la condición social de los padres, sobre todo si pertenecían a las clases más altas de la sociedad.

Los asientos bautismales proporcionan otro tipo de informaciones muy interesantes, ya que según el canon 1122.1, hay que incluir en el registro bautismal una anotación en el caso de que ese bautizado haya contraído matrimonio; asimismo, también debe incluirse nota sobre la recepción del sacramento de la confirmación que es el canon 895 o la adopción por esa persona de la condición de religioso según el canon 535.2 (García, 2007, p.5).

Palabra de origen griego que significa bañar, sumergir en agua, aunque debe entenderse, dentro de la fenomenología de la religión, en su sentido de purificación y vida nueva. Es el primero de los sacramentos de la iniciación cristiana, junto con la confirmación y la comunión. El Bautismo es el sacramento que nos incorpora a la Iglesia, nos comunica la vida de Hijos de Dios y nos limpia del pecado. En los primeros tiempos se bautizaba sólo los adultos pasando, a partir de los siglos VI y VII, a generalizarse el bautismo de los recién nacidos (Archivo Histórico Diocesano de Victoria, s.f.).

## **B. MATRIMONIO**

Conjunto de documentos que responden a un mismo tipo documental producido

por un mismo órgano en el ejercicio sus funciones, constituido por un antecedente y su consecuente, siempre y cuando no forme parte ya de otro fondo de Archivo (Archivo Histórico Diocesano de Victoria, s.f.).

También recibe el nombre de libro de casados o de velados. El canon 1121 recoge que tras la celebración del matrimonio el párroco "... debe anotar cuanto antes en el registro matrimonial los nombres de los cónyuges, del asistente y de los testigos, y el lugar y el día de la celebración, según el modo prescrito por la Conferencia Episcopal o por el Obispo diocesano". Este registro está íntimamente vinculado a la serie de "expedientes matrimoniales", en los que se recogen todos los antecedentes y pruebas que constaten que dos personas pueden contraer matrimonio, esto es, que no hay impedimentos sobre la celebración del enlace son expedientes ordinarios o, si los hubiesen, se incluiría además la documentación que demostrase la concesión de la correspondiente dispensa, que son expedientes extraordinarios (García, 2007, p.5).

El matrimonio es la unión de un hombre y una mujer, reconocida por el derecho e investida de ciertas consecuencias jurídicas. La definición anterior, corresponde a las tesis sostenidas por los tratadistas del siglo XX, que tenían una gran influencia del derecho romano e incluso del derecho canónico; a pesar de que son definiciones semejantes, en el derecho canónico al matrimonio se le considera como sacramento (Arellano, 2015).

### **C. DEFUNCIÓN**

Su existencia en las parroquias no siempre coincide con los de los bautismos y con los de matrimonios. Se indica el nombre y apellidos del finado y sus circunstancias personales, el motivo de la muerte, si testó, etc. En la documentación complementaria de estos libros aparecen los certificados de defunciones, aprobación de expedientes, etc. (Archivo Histórico Diocesano de Victoria, s.f.).

Las defunciones en sí no constituyen ningún sacramento, pero llevan implícitas la administración de uno: el de la extremaunción. El canon 1182 nos dice que "Una vez terminado el entierro, se ha de hacer la debida anotación en el libro de difuntos...". En su origen, en este libro sólo se inscribían a aquellos difuntos que habían incluido en su testamento el que se rezase por su alma y era muy habitual que en los asientos figurase la fórmula "mando lo siguiente" seguida del número de misas y el montante que se dejaba

para ello. En algunos libros, además, puede figurar la causa del fallecimiento y si recibió los últimos sacramentos (García, 2007, p.6).

Los libros de defunciones, también denominados de finados, no siempre coinciden en su arranque con los de bautizados y los matrimoniales. Estos tres tipos, junto con los de confirmaciones, integran el grupo de los libros parroquiales, o registros sacramentales, a los que pueden agregarse los libros borradores de partidas, expedientes matrimoniales y minutas de notas marginales.

Durante el siglo XVI, en un principio, estos libros recogían las mandas piadosas de quien habían otorgado testamento. Su estructura, como veremos en el apartado siguiente, es muy sencilla: data del sepelio y defunción, datos personales del difunto, circunstancia de haber testado y en ocasiones ante qué escribano.

En la centuria siguiente, a veces, en las actas de defunción de los “pobres de solemnidad” se deja constancia de que no han testado, por motivos obvios. También aparecen, en algunas grandes feligresías, a la par que los libros de adultos, los llamados “parvularios” donde se anotan específicamente los niños pequeños fallecidos.

Siguiendo al archivero Rubio Merino, la práctica de consignar en estos libros cláusulas testamentarias guardaba relación con su naturaleza, que más que obituarios, eran libros de testamentos en los que cada parroquia registraba las mandas pías y legados insertos en los testamentos, como consecuencia de los patronatos, capellanías o aniversarios en ellos instituidos. Esta teoría se puede comprobar viendo las propias partidas de defunción de la época moderna que estudiaremos en el siguiente apartado o leyendo los títulos que encabezan estos libros (Galende, 2011).

#### **D. CONFIRMACIÓN**

Segundo sacramento de la iniciación cristiana y de plena incorporación a la Iglesia, que nos infunde el Espíritu Santo, como a los discípulos en el día de Pentecostés, con el fin de hacernos más semejantes a Cristo y fortalecernos para el testimonio cristiano y para edificar la Iglesia. (Archivo Histórico Diocesano de Victoria, s.f.).

Administración del sacramento de la confirmación. No tenía carácter general en todas las



parroquias. Son relaciones nominales o listas de nombres de confirmados y sólo aparecen con la fecha, padrinos y, sólo en algunos casos, el nombre del obispo o arzobispo confirmante. (Archivo Histórico Diocesano de Victoria, s.f.).

La confirmación como sacramento, marca la evolución espiritual de un cristiano y siempre queda constancia de su impartición en el libro de bautismos. Pero algunas parroquias de cierta entidad pueden contar con un libro para registrar estas confirmaciones si así lo dispone el Obispo diocesano. En otros archivos parroquiales, lo que puede aparecer es un listado de todos aquellos que han recibido este sacramento y de sus padrinos, encabezado por el nombre del Obispo que lo impartió; este listado suele figurar anexo al libro de bautismos (García, 2007, p.6).

### **2.2.3. PROGRAMACIÓN EXTREMA**

Beck (2002), manifiesta que la programación extrema es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Jeffries, Anderson y Hendrickson (2000), Extreme Programming es una forma de desarrollar software basado en los valores de simplicidad, comunicación, feedback y coraje. Funciona haciendo trabajar a todo el equipo sobre un conjunto de prácticas sencillas, con suficiente feedback para habilitarle para ver dónde se encuentran y cómo aplicar dichas prácticas a cada situación particular.

### **FASES DE DESARROLLO DE LA PROGRAMACIÓN EXTREMA**

El ciclo de vida ideal de XP está formado de las siguientes fases: exploración, planificación, iteraciones y producción (Letelier y Penadés, 2006, p. 10).

#### **A. FASE DE EXPLORACIÓN**

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que

son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizan en el proyecto. Se prueba la tecnología y se explora las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología (Beck y Andres, 2004, p. 13).

## **B. FASE DE PLANIFICACIÓN**

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días. Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias generalmente valen de 1 a 3 puntos. Por otra parte, el equipo de desarrollo mantiene un registro de la “velocidad” de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración. La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación (Letelier y Penadés, 2006, p. 11).

## **C. FASE DE ITERACIONES**

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El plan de entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada el resto del proyecto. Al final de la última iteración el sistema estará listo para entrar en producción (Beck y Andres, 2004, p. 13).

## **D. FASE DE PUESTA EN PRODUCCIÓN**

La fase de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase. Es posible que se rebaje el tiempo que toma cada iteración, de tres a una semana. Las ideas que han sido propuestas y las sugerencias son documentadas para su posterior implementación (Letelier y Penadés, 2006, p. 12).

## **VARIABLES DE CONTROL PARA EQUILIBRAR LAS ACTIVIDADES EN UN PROYECTO DE XP EXITOSO**

### **A. TIEMPO**

Se debe dedicar tiempo para escuchar a los clientes, tiempo para diseñar, tiempo para codificar y tiempo para probar. Es difícil administrar el tiempo. La XP desafía la idea de que más tiempo le permitirá obtener los resultados que desea. Quizás el cliente preferiría que usted terminara a tiempo en lugar de extender la fecha límite para agregar otra función al sistema.

### **B. COSTO**

El costo puede usarse para equilibrar el proyecto. El tiempo extra tampoco ayuda mucho. Aumenta el costo, pero no siempre incrementa la productividad. Los programadores cansados son menos eficaces que los programadores alertas y tardan más tiempo para completar una tarea, y también cometen errores que requieren aún más tiempo para arreglarlos. A menudo es aconsejable invertir en herramientas como paquetes gráficos para comunicar a otros sus ideas sobre el proyecto, y las herramientas CASE pues contribuyen a acelerar los proyectos. Incluso el nuevo hardware podría ser un gasto redituable.

### **C. CALIDAD**

La calidad puede ajustarse tanto interna como externamente. La calidad interna involucra probar factores del software como la funcionalidad y la conformidad. Por lo general no es conveniente escatimar la calidad interior. En la calidad externa, o cómo el cliente percibe el sistema. Al cliente le interesa el desempeño. La filosofía extrema de XP permite sacrificar algunos de los aspectos de calidad externos. Para que el sistema sea

liberado a tiempo.

#### D. ALCANCE

El alcance se determina escuchando a los clientes y poniéndolos a redactar sus relatos. Los relatos deben ser breves y fáciles de comprender. Sería aún mejor si el analista pudiera determinar el tiempo y el dinero necesarios para satisfacer cada uno de estos relatos y establecer también su nivel de calidad; con el fin de mantener la calidad, manejar el costo y terminar el proyecto a tiempo, el analista de XP podría recurrir a ajustar el alcance del proyecto.

Tabla 2.1. Interacción entre las cuatro variables de Gestión de proyecto

VARIABLE	SI AUMENTA EN EXCESO...	SI SE REDUCE...
<b>ALCANCE</b>		Permite mejorar la calidad, siempre que resuelve el problema básico del cliente. También permite reducir plazo y coste. La herramienta más potente de gestión (*)
<b>TIEMPO</b>	Más puede mejorar calidad y alcance, pero en exceso en puede dañar, pues la mejor retroalimentación viene del sistema en producción.	Si poco, sufrirá la calidad e inmediatamente detrás el alcance, el tiempo y el coste.
<b>COSTO</b>	Más dinero puede engrasar el sistema, pero en exceso puede crear que lo resuelve.	Con dinero será imposible resolver los problemas del cliente.
<b>CALIDAD</b>	Insistir en mayor calidad permite conseguir plazos menores o hacer más en un tiempo dado. Efecto humano: se trabaja mejor si siente que se hace un buen trabajo.	Variable terrible de control. Se puede sacrificar para obtener ganancias a corto, pero los costes posteriormente son enormes (humanos, de negocio y técnicos)

Fuente: (Beck y Fowler, 2002).

## PRÁCTICAS ESENCIALES DEL ENFOQUE DE DESARROLLO DE XP



Figura 2.1. Prácticas esenciales del enfoque de desarrollo de XP (Kendall y Kendall, 2005).

### A. LIBERACIÓN LIMITADA

Para que el desarrollo de XP tenga éxito, los productos deben liberarse con rapidez. Esto significa que aun cuando los programadores no puedan implementar todas las características en una sola pieza de software, la versión debe liberarse de acuerdo con lo programado.

### B. SEMANA DE TRABAJO DE 40 HORAS

Esta práctica esencial de XP tiene como propósito motivar a los miembros del equipo a que laboren intensamente en el lugar de trabajo, y que tomen un periodo de descanso para que vuelvan al trabajo relajados y menos presionados, con capacidad de detectar los problemas y menos proclives a cometer errores.

### C. CLIENTE EN EL SITIO

La práctica esencial del cliente en el sitio llega al extremo al insistir en que un experto en el negocio debe trabajar en el sitio durante todo el proceso de desarrollo. Esta persona toma parte activa en el proceso, pues escribe los relatos del usuario, se comunica con los miembros del equipo y ayuda a establecer prioridades.

### D. PROGRAMACIÓN EN PAREJAS

Significa que dos programadores que eligen trabajar juntos hacen la programación, ejecutan las pruebas y conversan acerca de formas de hacer eficiente y eficazmente el trabajo. Al trabajar con otro programador puede clarificar su forma de pensar.

## **ROLES QUE SE DEBEN DESEMPEÑAR DURANTE EL PROCESO DE DESARROLLO DE XP**

Los siete roles son: programador, cliente, probador, rastreador, entrenador, consultor y gran jefe.



Figura 2.2. Roles que se deben desempeñar durante el proceso de desarrollo de XP (Kendall y Kendall, 2005).

### **A. PROGRAMADOR**

Se necesita contar con excelentes habilidades técnicas para programar, re factorizar y realizar pruebas unitarias al código que escriba. Además, necesita una buena disposición para abordar con sencillez los problemas más difíciles, aprender de otros, compartir el código y el diseño, y tener el valor para superar cualquier temor de incompetencia o fracaso al enfrentar nuevos problemas.

### **B. CLIENTE**

El cliente más adecuado para el equipo de XP es alguien que será usuario del sistema y que conoce la funcionalidad de negocios que éste requiere. El cliente debe

aprender a escribir relatos de usuario, aprender a escribir pruebas funcionales para las aplicaciones que generen los programadores, y tomar decisiones acertadas sobre las características esenciales del sistema e incluso sobre ajustes a la programación del proyecto y a las fechas de entrega.

#### **C. PROBADOR**

El programador también necesita comunicarse con el cliente sobre las pruebas de funcionamiento, realizar pruebas con regularidad, dar mantenimiento a las herramientas de prueba y elaborar informes precisos acerca de los resultados de las pruebas.

#### **D. RASTREADOR**

Éste da seguimiento al progreso general del grupo calculando el tiempo que toman sus tareas y el progreso general hacia sus metas. El rastreador realiza estimaciones de tiempo, pero también da retroalimentación acerca de las estimaciones del equipo. Los rastreadores también funcionan como memoria del equipo, al dar seguimiento a los resultados de todas las pruebas de funcionamiento.

#### **E. ENTRENADOR**

Los entrenadores son muy importantes. Ya que ellos conservan la calma cuando todo el equipo está asustado. Ellos moldean las situaciones de manera indirecta, y sólo de vez en cuando necesitan retirar con firmeza el mando a un programador errático, volver a encarrilarlo y devolverle las riendas otra vez.

#### **F. CONSULTOR**

El rol de un experto en consultoría técnica es muy singular. Lo que los equipos de desarrollo de XP esperan de un consultor es que éste les enseñe a resolver sus propios problemas. A medida que aprendan, renacerá en ellos la confianza, y cuando deje el equipo tal vez ellos utilicen o desechen la solución que les haya presentado, esto es común.

#### **G. GRAN JEFE O LÍDER**

El equipo espera que el gran jefe confíe en ellos, demuestre disposición para apegarse a los valores y principios a los que ellos se apeguen, y que tenga la capacidad

de señalar un error si el equipo se desvía del camino. El equipo requerirá mantenerse en comunicación con usted (incluso los pequeños cambios al diseño o las desviaciones de otras metas). Su tarea es conseguir que la comunicación fluya. Éste es un rol que exige una total convicción en el enfoque de XP y de que, si todos en el equipo se apegan a sus valores y principios básicos, es muy probable que logran algo que valga la pena.

## **HERRAMIENTAS DE PROGRAMACIÓN EXTREMA**

### **A. HISTORIAS DEL USUARIO**

Representan una breve descripción del comportamiento del sistema, emplea terminología del cliente sin lenguaje técnico, se realiza una por cada característica principal del sistema, se emplean para hacer estimaciones de tiempo y para el plan de lanzamientos, reemplazan un gran documento de requisitos y presiden la creación de las pruebas de aceptación.

Según Beck (1999), una historia de usuario es una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollo ágiles para la especificación de requerimientos (acompañadas de las discusiones con los usuarios y las pruebas de validación). Cada historia de usuario debe ser limitada, esta debería poderse escribir sobre una nota adhesiva pequeña. Dentro de la metodología XP las historias de usuario deben ser escritas por los clientes.

Las historias de usuario son una forma rápida de administrar los requerimientos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos. Las historias de usuario permiten responder rápidamente a los requerimientos cambiantes:

- a. Las historias de usuario tienen el mismo propósito que los casos de uso.
- b. Las escriben los propios clientes, tal y como ven ellos las necesidades del sistema.
- c. Las historias de usuario son similares al empleo de escenarios, con la excepción de que no se limitan a la descripción de la interfaz de usuario. También conducirán el proceso de creación de los test de aceptación (empleados para verificar que las historias de usuario han sido implementadas correctamente).
- d. Existen diferencias entre estas y la tradicional especificación de requisitos. La



principal diferencia es el nivel de detalle. Las historias de usuario solamente proporcionarán los detalles sobre la estimación del riesgo y cuánto tiempo conllevará la implementación de dicha historia de usuario.

Tabla 2.2. Modelo propuesto para una historia de usuario

<b>Historia de Usuario</b>	
<b>Número:</b>	<b>Nombre Historia de Usuario:</b>
<b>Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):</b>	
<b>Usuario:</b>	<b>Iteración Asignada:</b>
<b>Prioridad en Negocio:</b> (Alta / Media / Baja)	<b>Puntos Estimados:</b>
<b>Riesgo en Desarrollo:</b> (Alto / Medio / Bajo)	<b>Puntos Reales:</b>
<b>Descripción:</b>	
<b>Observaciones:</b>	

Fuente. (Porrás, 2010).

**Customer Story and Task Card** Blw Development / COLA

DATE: 3/19/98      TYPE OF ACTIVITY: NEW:  FIX:  ENHANCE:  FUNC. TEST:

STORY NUMBER: ~~1275~~ / 275      PRIORITY: USER:  TECH:

PRIOR REFERENCE: \_\_\_\_\_      RISK: \_\_\_\_\_ TECH ESTIMATE: \_\_\_\_\_

**TASK DESCRIPTION:**  
 SPLIT COLA: When the COLA rate chgs. in the middle of the Blw Pay Period, user will want to pay the 1<sup>st</sup> week of the pay period at the OLD COLA rate and the 2<sup>nd</sup> week of the Pay Period at the NEW COLA rate. Should occur automatically based on system design.  
 NOTES: on system design.  
 For the OT, we will run a m/frame program that will pay or calc the COLA on the 2<sup>nd</sup> week of OT. The plant currently retransmits the hours data for the 2<sup>nd</sup> week exclusively so that we can calc COLA. This will come into the Model as a "2144" COLA

**TASK TRACKING:** Gross Pay Adjustment. Create RM Boundary and Place in DEEnt Express COLA

Date	Status	To Do	Comments

Figura 2.3. Modelo propuesto para una historia de usuario (Beck y Fowler, 2002).

## B. TAREAS DE INGENIERÍA

Una historia de usuario se descompone en una o varias tareas de ingeniería, las

cuales describen las actividades que se realizarán en cada historia de usuario, así mismo las tareas de ingeniería se vinculan más al desarrollador, ya que permite tener un acercamiento con el código.

Tabla 2.3. Modelo propuesto para una tarea de ingeniería.

<b>Tarea de Ingeniería</b>	
<b>Número Tarea:</b>	<b>Historia de Usuario (Nro. y Nombre):</b>
<b>Nombre Tarea:</b>	
<b>Tipo de Tarea:</b> Desarrollo / Corrección / Mejora / Otra (especificar)	<b>Puntos Estimados:</b>
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b>	
<b>Descripción:</b>	

Fuente. (Porras, 2010).

**Customer Story and Task Card** BIW Development / COLA

DATE: 3/19/98      TYPE OF ACTIVITY: NEW:  FIX:  ENHANCE:  FUNC. TEST:

STORY NUMBER: ~~1275~~ / 1275      PRIORITY: USER:  TECH:

PRIOR REFERENCE: \_\_\_\_\_      RISK: \_\_\_\_\_ TECH ESTIMATE: \_\_\_\_\_

**TASK DESCRIPTION:**  
 SPLIT COLA: When the COLA rate chgs in the middle of the BIW Pay Period we will want to pay the 1<sup>st</sup> week of the pay period at the OLD COLA rate and the 2<sup>nd</sup> week of the Pay Period at the NEW COLA rate. Should occur automatically based on system design

**NOTES:**  
 For the OT, we will run a m/Frame program that will pay or calc the COLA on the 2<sup>nd</sup> week of OT. The plant currently retransmits the hours data for the 2<sup>nd</sup> week exclusively so that we can calc COLA. This will come into the Model as a "2144" COLA

**TASK TRACKING:** Gross Pay Adjustment, Create RM Boundary and Place in DE Ent Excess COLA

Date	Status	To Do	Comments	BIN

Figura 2.4. Modelo propuesto para una historia de usuario (Beck y Fowler, 2002).

### C. **TARJETAS CLASE - RESPONSABILIDAD – COLABORADOR (CRC)**

Estas tarjetas dividen en tres secciones que contienen la información del nombre de la clase, sus responsabilidades y sus colaboradores. En la siguiente figura se muestra cómo se distribuye esta información. “Cada tarjeta representa una clase en la programación orientada a objetos y define sus responsabilidades (lo que ha de hacer) y las colaboraciones con las otras clases (cómo se comunica con ellas)” (Beck, 1999).

Tabla 2.4. Modelo de tarjeta CRC

<b>Nombre de la clase</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>

### D. **CASO DE PRUEBAS DE ACEPTACIÓN**

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales, sobre el sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo, pues sería impresentable al cliente; sino que se realizan sobre el producto terminado e integrado o pudiera ser una versión del producto o una iteración funcional pactada previamente con el cliente.

La experiencia muestra que aún después del más cuidadoso proceso de pruebas por parte del desarrollador, quedan una serie de errores que sólo aparecen cuando el cliente comienza a usarlo. Los desarrolladores suelen llevar las manos a la cabeza y expresan: "Pero, ¿a quién se le ocurre usar así mi programa?", Sea como sea, el cliente siempre tiene razón. Decir que los requisitos no estaban claros, o que el manual es ambiguo puede salvar la cara; pero ciertamente no deja satisfecho al cliente. Alegar que el cliente es un inútil es otra tentación muy fuerte, que conviene reprimir.

Una prueba de aceptación puede ser desde un informal caso de prueba hasta la ejecución sistemática de una serie de pruebas bien planificadas. Las pruebas de aceptación pueden

tener lugar a lo largo de semanas o meses, descubriendo errores latentes o escondidos que pueden ir degradando el funcionamiento del sistema. Estas pruebas son muy importantes, porque definen pasar a fases del proyecto de despliegue y mantenimiento.

“La Prueba de Aceptación es una prueba formal conducida para determinar si un sistema satisface los criterios de aceptación y permite al cliente determinar si acepta o no el sistema” (Beck, 1999).

Tabla 2.5. Modelo propuesto para una prueba de aceptación

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b>	<b>Historia de Usuario (Nro. y Nombre):</b>
<b>Nombre:</b>	
<b>Descripción:</b>	
<b>Condiciones de Ejecución:</b>	
<b>Entrada / Pasos de ejecución:</b>	
<b>Resultado Esperado:</b>	
<b>Evaluación de la Prueba:</b>	

Fuente: (Porras, 2010).

## **ETAPAS DEL PROCESO DE DESARROLLO DE XP**

Hay cinco fases: exploración, planeación, iteraciones a la primera versión, puesta en producción y mantenimiento.

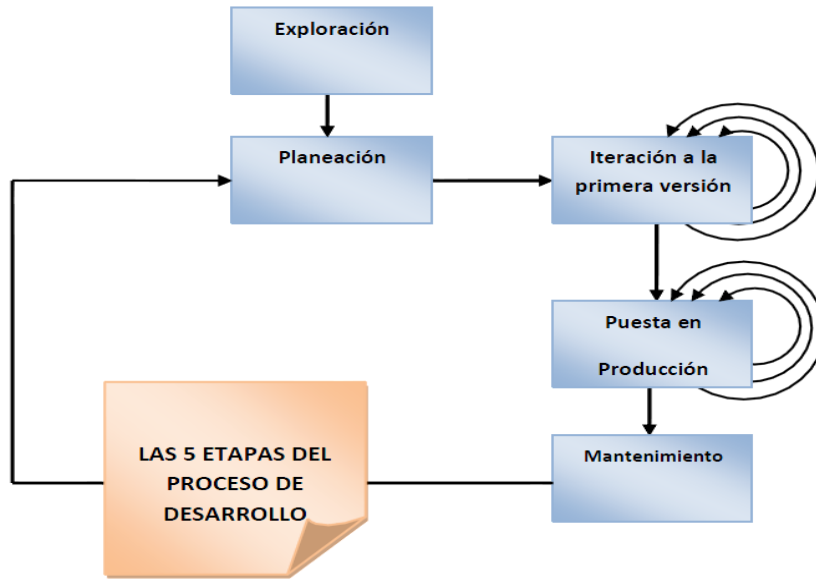


Figura 2.5. Etapas del proceso de desarrollo de XP (Kendall y Kendall, 2005).

#### A. **EXPLORACIÓN**

Se examinará su entorno, sosteniendo su convicción de que el problema puede y debe enfrentarse mediante programación extrema, conformará el equipo y valorará las habilidades de los miembros del mismo. Esta dura desde unas semanas hasta algunos meses y, también se ocupará de examinar las tecnologías potenciales que requerirá para construir el nuevo sistema. Durante esta etapa estimar el tiempo que tomarán el desarrollo de las diversas tareas. Los clientes también experimentarán con la escritura de relatos del usuario. El objetivo es lograr que el cliente refine lo suficiente un relato, para que usted pueda calcular con eficiencia la cantidad de tiempo que tomará construir la solución del sistema que está planeando.

#### B. **LA PLANEACIÓN**

La planeación podría tomar sólo algunos días. En esta etapa usted y sus clientes establecen una fecha de común acuerdo, que puede ir de dos meses a medio año a partir de la fecha actual, para la entrega de soluciones a los problemas de negocios más urgentes de los clientes.

#### C. **ITERACIONES A LA PRIMERA VERSIÓN**

Por lo general, estas iteraciones (ciclos de pruebas, retroalimentación y cambios) duran aproximadamente tres semanas. Tendrá que bosquejar toda la arquitectura del

sistema, aunque sólo sea un diseño preliminar. Una meta es realizar pruebas de funcionamiento escritas por el cliente al final de cada iteración. Al finalizar todas las iteraciones, el sistema está listo para pasar a la siguiente etapa.

#### **D. LA PUESTA EN PRODUCCIÓN**

Durante esta etapa se realiza diversas actividades. El ciclo de retroalimentación se acelera, de tal manera que en lugar de recibir retroalimentación para una iteración cada tres semanas, las revisiones del software se realizan en una semana. El producto se libera en esta etapa, aunque se puede mejorar incorporándole otras características.

#### **E. MANTENIMIENTO**

Una vez que se ha liberado el sistema, es necesario mantenerlo funcionando sin problemas. Se pueden agregar nuevas características, se pueden tomar en cuenta las sugerencias más arriesgadas del cliente y se pueden cambiar o incorporar nuevos miembros del equipo.

#### **2.2.4. APLICACIÓN WEB**

Una aplicación web, web-based application, es un tipo especial de aplicación cliente/servidor, donde el cliente: navegador, explorador o visualizador, y el servidor: el servidor web, y protocolo mediante el que se comunican (HTTP), están estandarizados y no son creados por el programador de aplicaciones (Luján, 2002, p. 48).

Para Seoane (2005), una aplicación web es un programa especialmente diseñado para ejecutarse dentro de un navegador web. Para ello se emplean tecnologías de tres capas, basándose en una arquitectura cliente-servidor: a) La primera capa reside en el ordenador del usuario, en el que se ejecutará la aplicación dentro del navegador web, se ocupa de la representación y obtención de datos, la generación de informes, gráficos; b) La segunda capa reside en el servidor de la lógica del negocio, que reside en el servidor, que además de preparar el entorno en el que se presenta la aplicación, se ocupa del procedimiento real de los datos, también es conocido como middleware; c) La tercera capa reside en el servidor de base de datos de la empresa, donde el servidor se ocupa de procesar las consultas que se efectúan desde el servidor de la lógica del negocio, de esta forma, devuelve los datos solicitados, disponiendo de módulos para crear y gestionar las bases de datos y los usuarios de las mismas.

### **2.2.5. APLICACIÓN MÓVIL**

Una aplicación móvil, es aquella que es ejecutada en una plataforma móvil desde cualquier lugar o zona geográfica utilizando la tecnología de conexión de datos (H, 3G, 4G, EDGE, entre otros), se hará uso de la señal en caso de necesitar conexión para guardar o consultar información (Quiroz, 2013, p. 39).

Una aplicación móvil, término proveniente del inglés App, que es una contracción de application con un tono informal, es un software que puede instalarse en un dispositivo móvil con la finalidad de extender sus funcionalidades, al igual que ocurre con la instalación de programas en computadoras de escritorio portátiles (Citado en Juárez, 2011, p.3).

### **2.2.6. PROGRAMACIÓN ORIENTADA A OBJETOS**

“La programación orientada a objetos o POO (OOP según sus siglas en inglés) es un paradigma de programación que usa objetos y sus interacciones, para diseñar aplicaciones y programas informáticos. Está basado en varias técnicas, incluyendo herencia, abstracción, polimorfismo y encapsulamiento. Su uso se popularizó a principios de la década de los años 1990. En la actualidad, existe variedad de lenguajes de programación que soportan la orientación a objetos” (Lowe, 2005).

“... Una clase es una descripción para producir objetos de esa clase o tipo. Una clase está formada por los métodos y los datos que definen las características comunes a todos los objetos de esa clase.”(Ceballos, 1993).

“Encapsulación significa encerrar, colocar cosas dentro de una capsula, en programación orientada a objetos la encapsulación se utiliza para ocultar información, así la interfaz de cada módulo muestra lo menos posible acerca del trabajo interno que realiza” (Campos, 2006).

Según Ceballos (1993), las características fundamentales de la OOP son: herencia, polimorfismo, abstracción y encapsulamiento.

“En la programación orientada a objetos se usa un mecanismo llamado herencia

(inheritance) para diseñar dos o más entidades que son diferentes pero que comparten muchas características comunes de las entidades” (Wu, 2008, p. 23).

La clase de quien se hereda se conoce como clase padre o clase ascendente o superclase, mientras que la clase que hereda se conoce como clase hija o clase descendiente o subclase. Esta clase hija a su vez puede convertirse en padre y así sucesivamente. A ésta descendencia se conoce como jerarquía de clases (Vásquez y Balta, 2008, p. 27).

El término polimorfismo expresa la posibilidad de que el mismo mensaje, enviado a objetos distintos, ejecute métodos distintos. Esto significa que podemos definir dentro de dos clases distintas dos operaciones con el mismo nombre y aspecto externo, pero con distintas implementaciones para cada clase (Noriega, 2007, p. 26).

Según Ceballos (1993), la abstracción “consiste en la generalización conceptual de los atributos y propiedades de un determinado conjunto de objetos. Precisamente la clave de la programación orientada a objetos está en abstraer los métodos y los datos comunes a un conjunto de objetos y almacenarlos en una clase”.

Piense que el encapsulamiento es como un envoltorio protector alrededor del código y los datos que se manipulan. El envoltorio define el comportamiento y protege el código y los datos para evitar que otro código acceda a ellos de manera arbitraria (Naughton, 1996, p. 17).

### **2.2.7. TECNOLOGÍAS DE INTERNET**

Internet es un conjunto de niveles de redes dispersas, que entre todas ellas conectan a millones de ordenadores, cuyos usuarios pueden intercambiar recursos informáticos, independientemente del ordenador que usen. Internet no es un sistema centralizado, no es una red, sino “red de redes”. Estas redes se conectan mediante líneas telefónicas convencionales, líneas de datos de alta velocidad, satélites, enlaces de microondas y fibra óptica (Jarabo y Elortegui, 1995, p. 1).

El protocolo clave utilizado por internet se llama, de manera apropiada, Protocolo Internet. Por lo general abreviado como IP, el protocolo específico, con minuciosidad, las reglas que definen los detalles de comunicación entre computadoras. Especifica



exactamente cómo se debe formar un paquete y como debe encaminar un ruteador cada paquete hacia su destino (Comer, 1995, p. 108).

#### **A. PROTOCOLO HTTP**

El protocolo HTTP forma parte de la familia de protocolos de comunicaciones Transmission Control Protocol/Internet Protocol (TCP/IP), que son empleados en Internet. Estos protocolos permiten la conexión de sistemas heterogéneos, lo que facilita el intercambio de información entre distintos ordenadores (Luján, 2002, p. 8).

El protocolo HTTP es un protocolo que funciona a nivel de aplicación según el modelo de referencia OSI, este protocolo se basa en el modelo cliente/servidor, es decir, consiste en un intercambio de mensaje entre dos dispositivos; a) El cliente, que es el que solicita servicios a un servidor. Su aplicación es lo que se conoce en Internet como navegadores, b) El servidor, que es el dispositivo que responde a las solicitudes de los clientes. Usualmente se les conoce como servidor Web, aunque en realidad son servidores de protocolo HTTP (Salavert, 2003, p. 142).

#### **B. PROTOCOLO TCP/IP**

El protocolo TCP (Transmission Control Protocol, Protocolo de control de transmisión) y el protocolo IP (Internet Protocol, Protocolo de Internet) controlan en envío y la recepción de información dentro de internet. El protocolo IP especifica el formato de los paquetes que se envían y reciben entre los routers y los sistemas terminales (Kurose y Ross, 2010).

“IP es un protocolo que proporciona mecanismos de interconexión entre redes de área local y TCP proporciona mecanismos de control de flujo y errores entre los extremos de la comunicación” (Barceló, Íñigo, Martí, Peig y Perramon, 2004, p. 71).

#### **C. PROGRAMACIÓN EN TRES CAPAS**

La programación por capas es una técnica de ingeniería de software propia de la programación por objetos, éstos se organizan principalmente en tres capas; a) Capa de presentación o frontera.- La presentación del programa ante el usuario, debe manejar interfaces que cumplan con el objetivo principal de este componente, el cual es facilitar al usuario la interacción con la aplicación. Para esto se utilizan patrones predefinidos para

cada tipo de aplicación y para cada necesidad del usuario. La interfaz debe ser amigable y fácil de utilizar, ya que el usuario final es el que se va a encargar de utilizar el sistema y de dar retroalimentación al equipo de desarrollo en caso de que haya algo que mejorar, b) Capa de lógica de negocio o control.- Es la llamada capa de reglas de negocios porque en esta se definen todas las reglas que se deben cumplir para una correcta ejecución del programa. Es aquí donde se encuentra toda la lógica del programa, así como las estructuras de datos y objetos encargados para la manipulación de los datos existentes, así como el procesamiento de la información ingresada o solicitada por el usuario en la capa de presentación. Recibe los datos que ingresó el usuario del sistema mediante la capa de presentación, luego los procesa y crea objetos según lo que se necesite hacer con estos datos, c) Capa de datos.- Es la encargada de realizar transacciones con bases de datos y con otros sistemas para obtener o ingresar información al sistema. El manejo de los datos debe realizar de forma tal que haya consistencia en los mismos, de tal forma los datos que se ingresan así como los que se extraen de las bases de datos, deben ser consistentes y precisos. Aquí se definen las consultas a realizar en la base de datos, tanto las consultas simples como consultas complejas para la generación de reportes más específicos (Vargas y Maltés, s.f.).

#### **D. INTERNET**

Stallings (2004) informa que, el Internet, es una red pública y global. Pública, porque puede ser utilizada por cualquier persona, a través del servicio provisto por alguno de los grandes proveedores de acceso a la misma y Global, debido a que conecta a miles de computadores alrededor del mundo. De acuerdo a sus características, Internet es un ambiente de trabajo sobre el cual debemos asumir, que no tenemos control alguno respecto de quiénes y cómo la utilizan. El usuario del otro lado, puede ser un potencial cliente, o un avezado atacante. En definitiva, no tendremos ninguna manera de conocer esta situación, al menos que estemos monitoreando sus acciones, en forma constante.

#### **2.2.8. SISTEMA DE GESTIÓN DE BASE DE DATOS**

Hansen y Hansen (2004) señalan que “Los sistemas de gestión de bases de datos (database management system, abreviado DBMS) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan”.

“Una base de datos es realmente un conjunto de objetos, de entidades. Estos objetos controlan la forma en la que se almacenan y gestionan los datos” (Mukhar y Lauiger, 2002).

“Un sistema de gestión de bases de datos relacional (SGBDR) da apoyo a la definición de datos mediante la estructura de los datos del modelo relacional, así como a la manipulación de estos datos con las operaciones del modelo; además, asegura que se satisfacen las reglas de integridad que el modelo relacional establece. El principal objetivo del modelo de datos relacional es facilitar que la base de datos sea percibida o vista por el usuario como una estructura lógica que consiste en un conjunto de relaciones y no como una estructura física de implementación. Esto ayuda a conseguir un alto grado de independencia de los datos. Un objetivo adicional del modelo es conseguir que esta estructura lógica con la que se percibe la base de datos sea simple y uniforme. Con el fin de proporcionar simplicidad y uniformidad, toda la información se representa de una única manera: mediante valores explícitos que contienen las relaciones (no se utilizan conceptos como por ejemplo apuntadores entre las relaciones). Con el mismo propósito, todos los valores de datos se consideran atómicos; es decir, no es posible descomponerlos” (Camps y Casillas, 2005).

“Una base de datos relacional es un conjunto estructurado de datos que pueden ser organizados de manera simple y eficiente por un manejador de dicha base. Las bases de datos actuales se denominan relacionales, ya que los datos pueden ser almacenados en diferentes tablas que facilitan su gestión y administración. Para ello y con el fin de estandarizar el acceso a las bases de datos se utiliza un lenguaje denominado SQL (Structured Query Language), que permite una interacción flexible, rápida e independiente de las aplicaciones a las bases de datos” (Jorba y Suppi, 2004).

### **2.2.9. POBLACIÓN Y MUESTRA**

#### **POBLACIÓN**

“La población se define como la totalidad del fenómeno a estudiar donde las unidades de población poseen una característica común la cual se estudia y da origen a los datos de la investigación” (Tamayo y Tamayo, 1997).

“Una población es un conjunto de todos los elementos que estamos estudiando, acerca de

los cuales intentamos sacar conclusiones” (Leviny y Rubin, 2004).

La población es el conjunto de todos los casos que concuerdan con una serie de especificaciones, podemos decir que la población es la totalidad del fenómeno a estudiar, en donde las unidades de población posee una característica común la cual estudia y da origen a los datos. (Hernández, 2000)

Bernal (2006) define que la población es el conjunto total de individuos, objetos o medidas que poseen algunas características comunes observables en un lugar y en un momento determinado. Cuando se vaya a llevar a cabo alguna investigación debe de tenerse en cuenta algunas características esenciales al seleccionarse la población bajo estudio. Entre éstas tenemos:

- a. Homogeneidad, que todos los miembros de la población tengan las mismas características según las variables que se vayan a considerar en el estudio o investigación.
- b. Tiempo, se refiere al período de tiempo donde se ubicaría la población de interés. Determinar si el estudio es del momento presente o si se va a estudiar a una población de cinco años atrás o si se van a entrevistar personas de diferentes generaciones.
- c. Espacio, se refiere al lugar donde se ubica la población de interés. Un estudio no puede ser muy abarcador y por falta de tiempo y recursos hay que limitarlo a un área o comunidad en específico.
- d. Cantidad, se refiere al tamaño de la población. El tamaño de la población es sumamente importante porque ello determina o afecta al tamaño de la muestra que se vaya a seleccionar, además que la falta de recursos y tiempo también nos limita la extensión de la población que se vaya a investigar.

#### **2.2.10. MUESTRA**

“La muestra se define como un subgrupo de la población. Para delimitar las características de la población” (Hernández, 2000).

Arias (1999) define a la muestra como una población o sea, un número de individuos, un objeto de los cuales es un elemento del universo o población, es decir, un conjunto de la población con la que se está trabajando por lo cual esta investigación se circunscribe a la cantidad de casos que se gestionan.

### **2.1.1 MUESTREO**

#### **MUESTREO POR JUICIO DE EXPERTOS**

También recibe el nombre de sesgado. El investigador selecciona los elementos que a su juicio son representativos, lo que exige un conocimiento previo de la población que se investiga (Tamayo y Tamayo, 1997).

Según Bavaresco (2006), los sujetos de una investigación específica, son seleccionados para el estudio porque son más fáciles de reclutar y el investigador no está considerando las características de inclusión de los sujetos representativos de toda la población.

## **CAPÍTULO III METODOLOGÍA DE LA INVESTIGACIÓN**

### **3.1. TIPO DE INVESTIGACIÓN**

El estudio es observacional porque no se interviene sobre la variable de estudio fondo documental, según la planificación de la medición es retrospectiva porque los datos no son generados por el investigador, es transversal porque se recolecta datos para la variable en un solo momento y descriptiva porque se describirá mediante las variables descriptivas; bautismo, matrimonio, defunción y confirmación.

### **3.2. NIVEL DE INVESTIGACIÓN**

El nivel de estudio es descriptivo, porque tiene una sola variable de investigación denominada fondo documental, y el desarrollo del software permitirá describir las variables descriptivas.

### **3.3. DISEÑO DE LA INVESTIGACIÓN**

El diseño de estudio es observacional, transversal y retrospectivo, porque recolectamos información del fondo documental sobre el: matrimonio, bautizo, confirmación y defunción; para la recolección de datos se aplica la técnica de análisis documental y el instrumento registro, a fin de obtener información para las historias de

usuario del prototipo de software, procesado mediante la programación extrema, la funcionalidad del software permitirá describir el fondo documental de la Arquidiócesis de Ayacucho.

### **3.4. POBLACIÓN Y MUESTRA**

#### **POBLACIÓN**

La población está compuesta por; 850 matrimonios, 1200 bautizos, 820 confirmaciones y 480 defunciones, de los años 1900 a 1940 de la Arquidiócesis de Ayacucho.

#### **MUESTRA**

Se ha tomado una muestra por juicio de expertos de los fondos documentales siendo: 30 bautizos, 30 matrimonios, 30 defunciones y 30 confirmaciones, de los años 1900 a 1940 de la Arquidiócesis de Ayacucho.

### **3.5. VARIABLES E INDICADORES**

#### **DEFINICIÓN CONCEPTUAL DE LAS VARIABLES**

##### **VARIABLE DE INTERÉS**

###### **Fondo documental**

Se entiende por fondo documental, la agrupación orgánica de documentos generados por una institución en el ejercicio de sus funciones, que constituyen la expresión del conjunto de las actividades sociales desarrolladas.

##### **VARIABLES DESCRIPTIVAS**

###### **Bautismo**

El bautismo anotado en el libro de bautizo de la arquidiócesis presenta los siguientes datos: nombre del bautizado, haciendo mención del ministro, los padres del bautizado, padrinos, testigos si los hubo, lugar y fecha del bautizo, indicando asimismo, la fecha y lugar del nacimiento del bautizado.

###### **Matrimonio**

Es la unión de un hombre y una mujer, registrado en el libro de matrimonio de la arquidiócesis, reconocida por el derecho e investida de consecuencias jurídicas, siendo

los datos registrados: nombre de la parroquia, fecha, nombre del párroco, nombre del contrayente, nombre de la contrayente, nombre del padre y madre del contrayente, nombre del padre y madre de la contrayente, nombre de los padrinos.

### **Defunción**

La defunción es un documento legal que certifica el fallecimiento de una persona, registrado en la partida de defunciones de la arquidiócesis, indicando: nombre y apellidos, fecha de defunción, edad, nombre de los padres del difunto, lugar de procedencia del difunto, causa del fallecimiento.

### **Confirmación**

Como segundo sacramento de la iniciación cristiana y de plena incorporación a la Iglesia, registrado en el libro de confirmaciones de la arquidiócesis, con el fin de hacernos más semejantes a Cristo y fortalecernos para el testimonio cristiano y para edificar la Iglesia, sobre este sacramento se ha registrado: número de confirmado, fecha de bautismo, lugar del bautismo, nombre del confirmado, nombre de los padres, nombre de los padrinos, ministerio de la confirmación y fecha.

## **DEFINICIÓN OPERACIONAL DE LA VARIABLE**

### **VARIABLE DE INTERÉS**

**X:** Fondo documental.

### **VARIABLES DESCRIPTIVAS**

**X1:** Bautismo.

**X2:** Matrimonio.

**X3:** Defunción.

**X4:** Confirmación.

## **OPERACIONALIZACIÓN DE LA VARIABLE**

La operacionalización de la variable de interés se muestra en el anexo 1.

## **3.6. TÉCNICAS E INSTRUMENTOS PARA RECOLECCIÓN DE INFORMACIÓN**

## TÉCNICA PARA RECOLECTAR LA INFORMACIÓN

La técnica que se utilizó para levantar la información fue el análisis documental, para el estudio del fondo documental de la arquidiócesis de Ayacucho.

## INSTRUMENTO PARA RECOLECTAR LA INFORMACIÓN

El instrumento utilizado fue el registro, para levantar información sobre los fondos documentales como: bautizos, matrimonios, defunciones y confirmaciones del Archivo Arzobispal de Ayacucho.

### 3.7. HERRAMIENTAS PARA EL TRATAMIENTO DE DATOS

Las herramientas tecnológicas utilizadas, fueron seleccionadas en función a limitaciones existentes para el proyecto, como; aplicación pequeña, existe limitados recursos humanos, infraestructura y financiamiento para el desarrollo del software web.

Tabla 3.1. Tecnologías para el tratamiento de datos

SOFTWARE	FABRICANTE	SERVICIO
Windows 10 Home	Microsoft Corporation	Es la versión del sistema operativo Windows para escritorio, ofrece el marco para instalar las herramientas de desarrollo.
Ubuntu 18.04	Canonical	Ubuntu es un sistema operativo basado en GNU/Linux y que se distribuye como software libre.
PostgreSQL V 9.6	PostgreSQL Global Development Group	PostgreSQL es un Sistema de gestión de bases de datos relacional orientado a objetos y libre
Pycharm 2018.3	JetBrains	PyCharm es un entorno de desarrollo integrado (IDE) utilizado en la programación de computadoras, específicamente para el lenguaje Python.
Apache 2.4	Apache Software Foundation	Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y



		otras, que implementa el protocolo HTTP/1.1.
StartUml 2.8	StartUml	StarUML es una herramienta para el modelamiento de software basado en los estándares UML (Unified Modeling Language) y MDA (Model Driven Architecture)
Python 3.6	Python Software Foundation	Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.
Django 2.1	Django Software Foundation	Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo– vista– controlador.

### 3.8. TÉCNICAS PARA APLICAR PROGRAMACIÓN EXTREMA

Revisado el marco teórico desarrollado en el capítulo II, sección 2.2.3, formulamos el proceso, que considera las fases para desarrollar el software usando XP, como se muestra en las tablas 3.2 a 3.4.

Tabla 3.2. Fase de Exploración

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLES
Escribir historias de usuario	Historia de usuario	Describir brevemente la historia de usuario con la regla del negocio (lo que el sistema debe hacer) Dividir historias de usuario grandes	Cliente
Probar las tecnologías a utilizar	Arquitectura técnica inicial	Explorar posibilidades de uso de tecnologías Probar el rendimiento de las tecnologías Definir las tecnologías a usar	Cliente Programador Entrenador

Estimar esfuerzo para historia de usuario	Plan de alto nivel	Conocer previamente la historia de usuario Hacer una implementación rápida de historia de usuario Estimar esfuerzo (semana) para desarrollar la historia de usuario	Programador
---	--------------------	---	-------------

Fuente: (Porras, 2010)

Tabla 3.3. Fase de Planificación

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLES
Rescribir las historias de usuario	Historia de usuario	Describir detalladamente la historia de usuario con la regla de negocio	Cliente
Formular el plan de versiones	Plan de versión (una iteración)	Introducir nuevos requisitos del software Definir prioridad para cada historia de usuario por necesidad del negocio	Cliente
		Utilizar técnicas de elaboración del plan de alto nivel Estimar y asignar esfuerzo (semana) para cada historia de usuario en función a tiempo para planear, diseñar, implementar y probar Estimar y asignar riesgo a cada historia de usuario en función a situación que afecta la estimación del esfuerzo Actualizar tarjeta de historia de usuario	Programador

Fuente: (Porras, 2010)

Tabla 3.4. Fase de Iteración

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLES
Definir la arquitectura técnica	Arquitectura técnica	Actualizar la arquitectura técnica inicial Usar características del negocio Utilizar arquitectura por capas Integrar frameworks	Cliente Programador Entrenador
Escribir tareas de ingeniería	Tarea de ingeniería	Dividir cada historia de usuario en tareas, describir usando reglas del negocio cada tarea de ingeniería	Cliente programador
Formular el plan de	Plan de iteración	Estimar y asignar esfuerzo para desarrollar una tarea de	Programador

iteraciones		ingeniería	
		Asignar una tarea de ingeniería al programador	Entrenador Programador
		Utilizar el plan de versión Actualizar el plan con tareas de ingeniería de la siguiente iteración Actualizar las historias de usuario Actualizar el plan con tareas no concluidas Actualizar las tarjetas de tarea de ingeniería	
Implementar las interfaces	GUI	Diseñar con precisión la GUI relacionada a cada historia de usuario Generar código para la interface usando herramienta	Cliente Programador
Escribir tarjetas CRC para cada tarea de ingeniería	Tarjeta CRC	Diseñar para una tarea de ingeniería de forma simple Rediseñar por falla de prueba de aceptación una tarea Identificar responsabilidades Identificar colaboración Identificar Atributos	Cliente Programador
Implementar la base de datos física	Base de datos física	Escribir script usando tarjeta CRC Ejecutar script usando DBMS	Programador
Implementar código para clases entidad	Código fuente	Escribir código fuente o generar con una herramienta usando tarjetas CRC	Programador
Crear pruebas unitarias para las clases control	Prueba unitaria	Escribir código fuente para una prueba unitaria, usando una herramienta	Programador
Implementar código fuente	Código fuente	Codificar una tarea de ingeniería Hacer refactoring Mover programadores	Programador Supervisor
Ejecutar pruebas unitarias	Reporte de prueba unitaria	Ejecutar el módulo de cada prueba unitaria Modificar código fuente si la prueba unitaria muestra resultado incorrecto	Programador
Realizar integración continua	Código fuente	Integrar las tareas para una historia de usuario Mantener sistema integrado	Programador

		todo el tiempo	
Ejecutar pruebas de integración para una historia de usuario	Reporte pruebas de integración	Integrar continuamente al concluir las tareas de una historia de usuario Verificar que las pruebas de integración pasan al 100%	Programador
Ejecutar pruebas de aceptación	Reporte de pruebas de aceptación	Correr la última versión de una iteración Utilizar los casos de prueba de aceptación	Cliente Encargado de pruebas

Fuente: (Porras, 2010)

## CAPÍTULO IV

### ANÁLISIS Y RESULTADOS DE LA INVESTIGACIÓN

#### 4.1. ARTEFACTOS DEL SOFTWARE APLICANDO PROGRAMACIÓN EXTREMA

Aplicando las técnicas del proceso XP, resumidas en las tablas 3.2 a 3.4, obtenemos los artefactos como tablas y figuras para el modelado e implementación del software que administrar el fondo documental de la arquidiócesis de Ayacucho.

##### 4.1.1. FASE DE EXPLORACIÓN

Según el procedimiento desarrollado en la tabla 3.2, para el ciclo de vida de un proyecto programación extrema. Obtenemos, las historias de usuario, arquitectura inicial y el plan de alto nivel, que se muestra continuación.

Tabla 4.1. Historias de usuario

Nº	HISTORIA DE USUARIO	DESCRIPCIÓN
1	Mantener datos de cada <b>bautismo</b> (Web y móvil)	El <b>archivero</b> debe actualizar los datos de cada <b>bautismo</b> realizado y custodiado en el Archivo Arzobispal de Ayacucho.
2	Mantener datos de cada <b>matrimonio</b> (Web y móvil)	El archivero debe actualizar los datos de cada <b>matrimonio</b> realizado y custodiado en el Archivo Arzobispal de Ayacucho.
3	Mantener datos de cada <b>defunción</b> (Web y móvil)	El archivero debe actualizar los datos de cada <b>defunción</b> registrado y custodiado en el Archivo Arzobispal de Ayacucho.
4	Mantener datos de cada <b>confirmación</b> (Web y móvil)	El archivero debe actualizar los datos de cada <b>confirmación</b> realizado y custodiado en el Archivo Arzobispal de Ayacucho.

5	Almacenar la <b>imagen</b> de un fondo documental (Web)	El <b>archivero</b> debe almacenar la <b>imagen</b> de un <b>fondo documental</b> custodiado en el Archivo Arzobispal de Ayacucho.
6	Consultar los bautismos (Web y móvil)	El <b>ciudadano</b> debe ser capaz de consultar los bautismos realizados y custodiados en el Archivo Arzobispal de Ayacucho.
7	Consultar los matrimonios (Web y móvil)	El ciudadano debe ser capaz de consultar los matrimonios realizados y custodiados en el Archivo Arzobispal de Ayacucho.
8	Consultar las defunciones (Web y móvil)	El ciudadano debe ser capaz de consultar las defunciones registradas y custodiadas en el Archivo Arzobispal de Ayacucho.
9	Consultar las confirmaciones (Web y móvil)	El ciudadano debe ser capaz de consultar las confirmaciones realizadas y custodiadas en el Archivo Arzobispal de Ayacucho.
10	Consultar la imagen de un fondo documental (Web y móvil)	El ciudadano puede consultar la imagen de un fondo documental custodiado en el Archivo Arzobispal de Ayacucho.
11	Iniciar sesión de usuarios a la aplicación (Web y Móvil)	Los usuarios inician sesión con su clave y nombre de usuario.

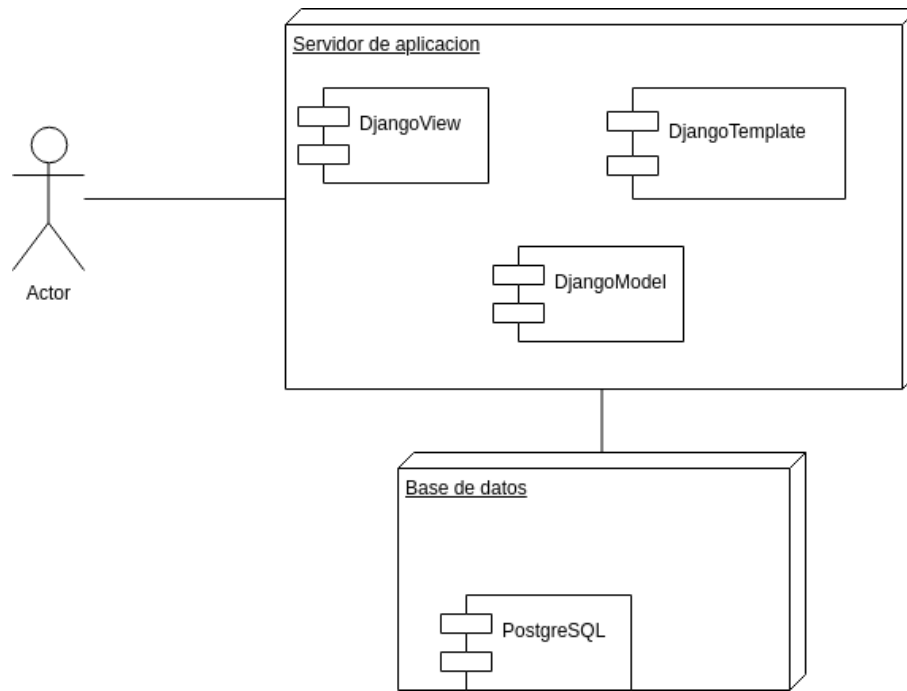


Figura 4.1. Arquitectura Técnica Inicial

Tabla 4.2. Plan de alto nivel

N°	HISTORIAS DE USUARIO	ESFUERZO (DÍAS)
1	Mantener datos de cada bautismo (Web y móvil)	3.0
2	Mantener datos de cada matrimonio (Web y móvil)	3.0
3	Mantener datos de cada defunción (Web y móvil)	3.0
4	Mantener datos de cada confirmación (Web y móvil)	3.0
5	Almacenar la imagen de un fondo documental (Web)	2.0
6	Consultar los bautismos (Web y móvil)	2.0
7	Consultar los matrimonios (Web y móvil)	2.0
8	Consultar las defunciones (Web y móvil)	2.0
9	Consultar las confirmaciones (Web y móvil)	2.0
10	Consultar la imagen de un fondo documental (Web y móvil)	2.0
11	Iniciar sesión de usuarios a la aplicación (Web y Móvil)	2.0

#### 4.1.2 FASE DE PLANIFICACIÓN

Aplicamos la técnica para la fase de planificación, presentada en la tabla 3.3 del capítulo III, obteniendo los artefactos, historias de usuario en detalle y el plan de versión (primera iteración), mostrados a continuación.

Tabla 4.3. Historia usuario. Mantener datos de cada bautismo.

<b>Historia de Usuario</b>	
<b>Número:</b> 1	<b>Usuario:</b> Archivero
<b>Nombre historia:</b> Mantener datos de cada bautismo	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Programador - Asistente	
<b>Descripción:</b> El <b>archivero</b> registra los datos de cada bautismo. Los datos ingresados se almacenan en la base de datos. El <b>archivero</b> consulta los datos de cada bautismo. Los datos se muestran en la pantalla de la computadora. El <b>archivero</b> modifica los datos de cada bautismo. Los datos modificados se almacenan en la base de datos. El <b>archivero</b> elimina los datos de cada bautismo. Los datos se eliminan de la base de datos.	
Observaciones: Aplicación Web y móvil	

Tabla 4.4. Historia usuario. Mantener datos de cada matrimonio.

<b>Historia de Usuario</b>	
<b>Número:</b> 2	<b>Usuario:</b> Archivero
<b>Nombre historia:</b> Mantener datos de cada matrimonio	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Programador - Asistente	
<b>Descripción:</b> El <b>archivero</b> registra los datos de cada matrimonio. Los datos ingresados se almacenan en la base de datos.	



El <b>archivero</b> consulta los datos de cada matrimonio. Los datos se muestran en la pantalla de la computadora.
El <b>archivero</b> modifica los datos de cada matrimonio. Los datos modificados se almacenan en la base de datos.
El <b>archivero</b> elimina los datos de cada matrimonio. Los datos se eliminan de la base de datos.
Observaciones: Aplicación web y móvil

Tabla 4.5. Historia usuario. Mantener datos de cada defunción.

Historia de Usuario	
<b>Número:</b> 3	<b>Usuario:</b> Archivero
<b>Nombre historia:</b> Mantener datos de cada defunción	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Programador - Asistente	
<p><b>Descripción:</b></p> <p>El <b>archivero</b> registra los datos de cada defunción. Los datos ingresados se almacenan en la base de datos.</p> <p>El <b>archivero</b> consulta los datos de cada defunción. Los datos se muestran en la pantalla de la computadora.</p> <p>El <b>archivero</b> modifica los datos de cada defunción. Los datos modificados se almacenan en la base de datos.</p> <p>El <b>archivero</b> elimina los datos de cada defunción. Los datos se eliminan de la base de datos.</p>	
Observaciones: Aplicación web y móvil	

Tabla 4.6. Historia usuario. Mantener datos de cada confirmación.

Historia de Usuario	
<b>Número:</b> 3	<b>Usuario:</b> Archivero
<b>Nombre historia:</b> Mantener datos de cada confirmación	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Programador - Asistente	

<p><b>Descripción:</b></p> <p>El <b>archivero</b> registra los datos de cada confirmación. Los datos ingresados se almacenan en la base de datos.</p> <p>El <b>archivero</b> consulta los datos de cada confirmación. Los datos se muestran en la pantalla de la computadora.</p> <p>El <b>archivero</b> modifica los datos de cada confirmación. Los datos modificados se almacenan en la base de datos.</p> <p>El <b>archivero</b> elimina los datos de cada confirmación. Los datos se eliminan de la base de datos.</p>
Observaciones: Aplicación web y móvil

Tabla 4.7. Historia usuario. Almacenar la imagen de un fondo documental.

Historia de Usuario	
<b>Número:</b> 5	<b>Usuario:</b> Archivero
<b>Nombre historia:</b> Almacenar la imagen de un fondo documental.	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador - Asistente	
<p><b>Descripción:</b></p> <p>El <b>archivero</b> actualiza la base de datos con la imagen de un fondo documental, siendo los fondos; bautismo, matrimonio, defunción y confirmación.</p>	
Observaciones: Aplicación web	

Tabla 4.8. Historia usuario. Consultar los bautismos.

Historia de Usuario	
<b>Número:</b> 6	<b>Usuario:</b> Usuario
<b>Nombre historia:</b> Consultar los bautismos.	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El usuario (archivero o ciudadano) realiza una consulta usando su computadora o teléfono móvil de los bautismos realizados. Se consulta desde la base de datos web.</p>	

Observaciones: Aplicación web y móvil
---------------------------------------

Tabla 4.9. Historia usuario. Consultar los matrimonios.

Historia de Usuario	
<b>Número:</b> 7	<b>Usuario:</b> Usuario
<b>Nombre historia:</b> Consultar los matrimonios.	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador – Asistente	
<b>Descripción:</b> El usuario (archivero o ciudadano) realiza una consulta usando su computadora o teléfono móvil de los matrimonios realizados. Se consulta desde la base de datos web.	
Observaciones: Aplicación web y móvil	

Tabla 4.10. Historia usuario. Consultar las defunciones.

Historia de Usuario	
<b>Número:</b> 8	<b>Usuario:</b> Usuario
<b>Nombre historia:</b> Consultar las defunciones.	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador – Asistente	
<b>Descripción:</b> El usuario (archivero o ciudadano) realiza una consulta usando su computadora o teléfono móvil de las defunciones registradas. Se consulta desde la base de datos web.	
Observaciones: Aplicación web y móvil	

Tabla 4.11. Historia usuario. Consultar las confirmaciones.

Historia de Usuario	
<b>Número:</b> 9	<b>Usuario:</b> Usuario
<b>Nombre historia:</b> Consultar las confirmaciones.	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador – Asistente	

<b>Descripción:</b> El usuario (archivero o ciudadano) realiza una consulta usando su computadora o teléfono móvil de las confirmaciones registradas. Se consulta desde la base de datos web.
Observaciones: Aplicación web y móvil

Tabla 4.12. Historia usuario. Consultar la imagen de un fondo documental.

<b>Historia de Usuario</b>	
<b>Número:</b> 10	<b>Usuario:</b> Usuario
<b>Nombre historia:</b> Consultar la imagen de un fondo documental	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador - Asistente	
<b>Descripción:</b> El usuario (archivero o ciudadano) realiza una consulta usando su computadora o teléfono móvil del fondo documental. Se consulta desde la base de datos web.	
Observaciones: Aplicación web y móvil	

Tabla 4.13. Historia usuario. Iniciar sesión de usuarios a la aplicación.

<b>Historia de Usuario</b>	
<b>Número:</b> 11	<b>Usuario:</b> Usuario
<b>Nombre historia:</b> Iniciar sesión de usuarios a la aplicación	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Programador - Asistente	
<b>Descripción:</b> El usuario (archivero o ciudadano) inicia sesión desde su computadora o teléfono móvil, ingresando su usuario y clave de acceso.	
Observaciones: Aplicación web y móvil	

Tabla 4.14. Plan de Versión

Nº	HISTORIA DE USUARIO	PRIORIDAD	RIESGO	ESFUERZO DÍAS	ITERACIÓN
----	---------------------	-----------	--------	------------------	-----------

1	Mantener datos de cada bautismo.	Alto	Medio	3.0	1
2	Mantener datos de cada matrimonio.	Alto	Medio	3.0	1
3	Mantener datos de cada defunción.	Alto	Medio	3.0	1
4	Mantener datos de cada confirmación.	Alto	Medio	3.0	1
5	Almacenar la imagen de un fondo documental.	Alto	Medio	2.0	1
6	Consultar los bautismos.	Medio	Medio	2.0	2
7	Consultar los matrimonios.	Medio	Medio	2.0	2
8	Consultar las defunciones.	Medio	Medio	2.0	2
9	Consultar las confirmaciones.	Medio	Medio	2.0	2
10	Consultar la imagen de un fondo documental.	Medio	Medio	2.0	2
11	Iniciar sesión de usuarios a la aplicación (Web y Móvil)	Medio	Medio	2.0	2

#### 4.1.3 FASE DE ITERACIÓN

De acuerdo a la técnica desarrollada en la tabla 3.4, obtenemos los entregables; arquitectura técnica, tareas de ingeniería, plan de iteración, casos de prueba de aceptación, GUI, tarjetas CRC, base de datos física, código fuente para clases entidad, código fuente para tarea de ingeniería, reporte de aceptación, presentados a continuación.

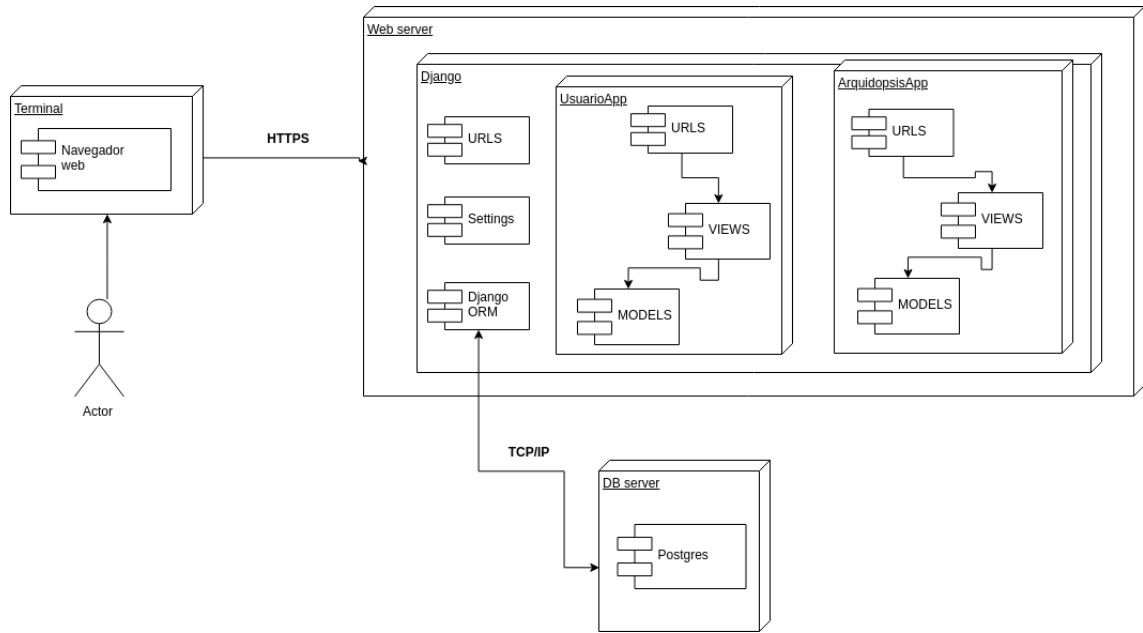


Figura 4.2. Arquitectura técnica final de la Aplicación web. Diagrama de despliegue

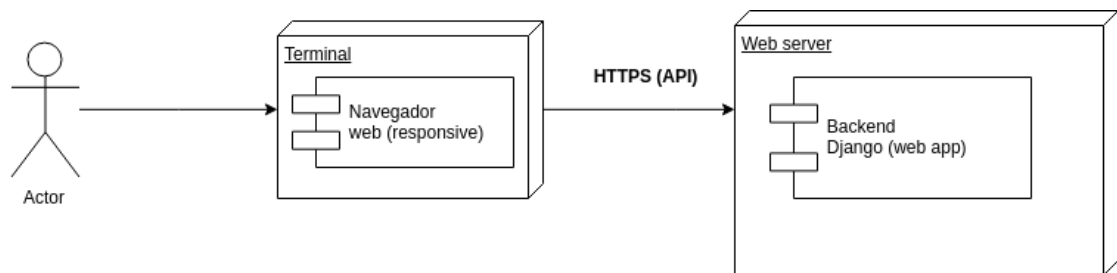


Figura 4.3. Arquitectura técnica final de la Aplicación Móvil. Diagrama de despliegue

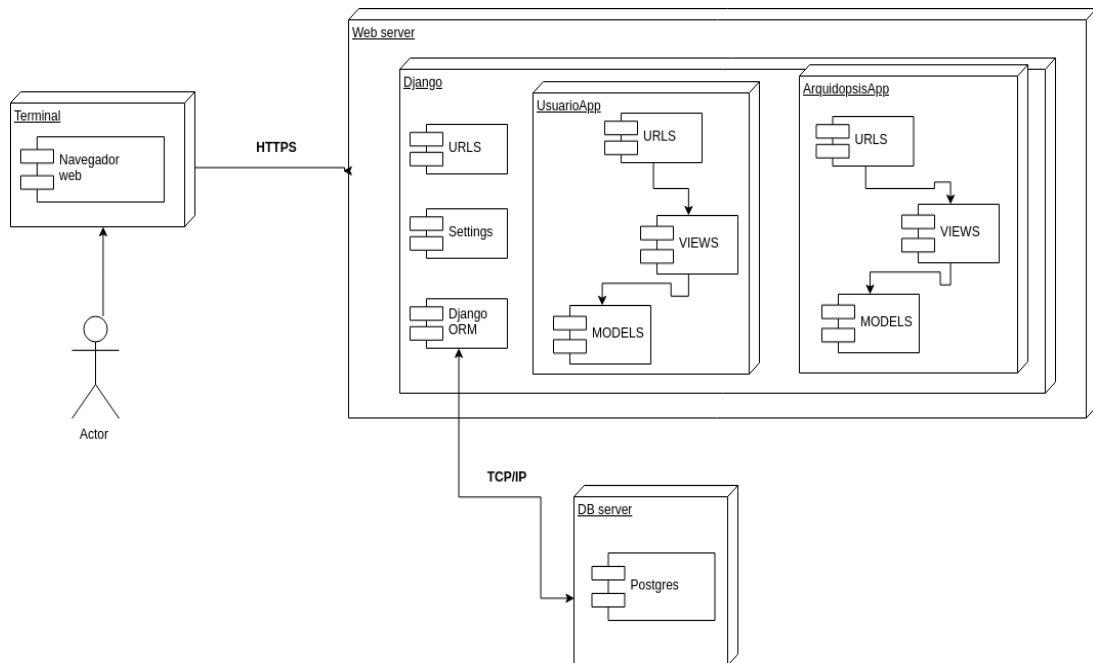


Figura 4.4. Arquitectura técnica final de la Aplicación web. Diagrama de componentes

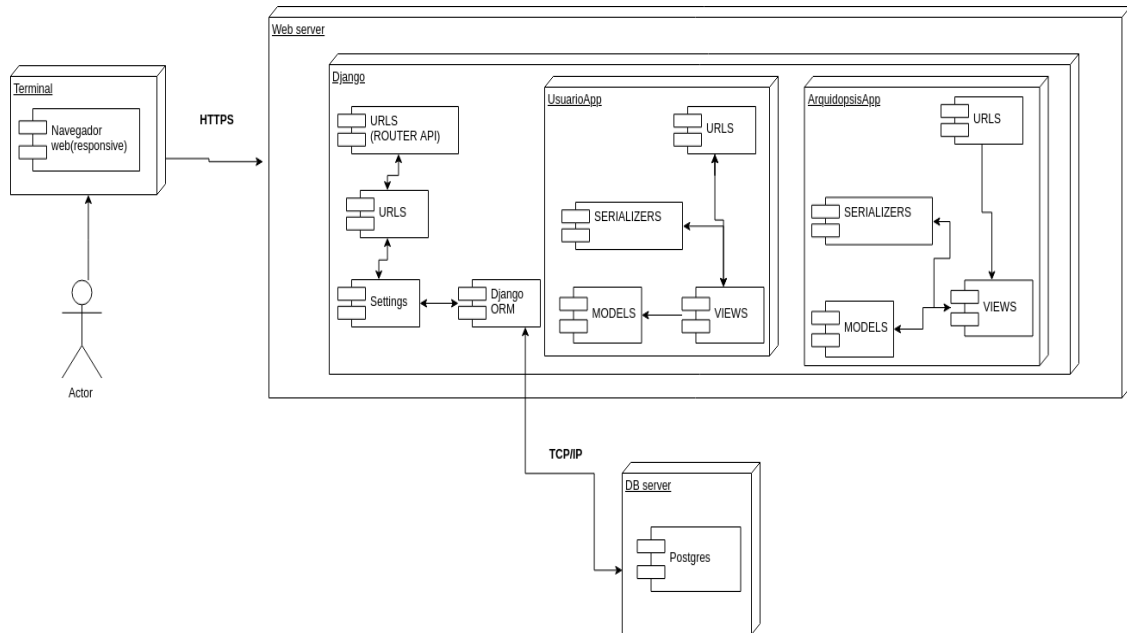


Figura 4.5. Arquitectura técnica final Aplicación Móvil. Diagrama de componentes

Tabla 4.15. Tarea de Ingeniería. Registrar bautismo.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 1	<b>Numero de historia de usuario:</b> 1
<b>Nombre tarea:</b> Registrar bautismo	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 02/05/2019	<b>Fecha fin:</b> 02/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<b>Descripción:</b> El archivero usa la interfaz “registrar bautismo”. Ingresa los atributos nombre del bautizado como: ministro, padres del bautizado, padrinos, testigos, lugar y fecha del bautizo, fecha y lugar de nacimiento del bautizado. Hace clic en el botón “registrar”. Los atributos ingresados se almacenan en la base de datos. La aplicación web muestra el mensaje “grabación correcta”.	
<b>Observaciones:</b> Cuando no se graba correctamente se muestra el mensaje “Error de registro”.	

Tabla 4.16. Tarea de Ingeniería. Modificar bautismo.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 2	<b>Numero de historia de usuario:</b> 1
<b>Nombre tarea:</b> Modificar bautismo	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 04/05/2019	<b>Fecha fin:</b> 04/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “modificar bautismo”, para ingresar el atributo clave del bautismo.</p> <p>Hace clic en el botón “buscar” y los atributos como: ministro, padres del bautizado, padrinos, testigos, lugar y fecha del bautizo, fecha y lugar de nacimiento del bautizado; se muestran en la interfaz “modificar bautismo”.</p> <p>Modifica algunos atributos hace clic en el botón “modificar”.</p> <p>Los atributos modificados se almacenan en la base de datos.</p> <p>La aplicación web muestra el mensaje “modificación correcta”.</p>	
<p>Observaciones: Cuando no se modifica correctamente se muestra el mensaje “Error de modificación”.</p>	

Tabla 4.17. Tarea de Ingeniería. Eliminar bautismo.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 3	<b>Numero de historia de usuario:</b> 1
<b>Nombre tarea:</b> Eliminar bautismo	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 05/05/2019	<b>Fecha fin:</b> 05/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “eliminar bautismo”, para ingresar el atributo clave de un bautismo.</p> <p>Hace clic en el botón “buscar”, los atributos como: ministro, padres del bautizado, padrinos, testigos, lugar y fecha del bautizo, fecha y lugar de nacimiento del bautizado se muestran en la interfaz “eliminar bautismo”.</p>	



El archivero hace clic en el botón “eliminar”.
El bautismo se elimina de la base de datos.
La aplicación web muestra el mensaje “eliminación correcta”.
Observaciones: Cuando no se elimina correctamente se muestra el mensaje “Error de eliminación”.

Tabla 4.18. Tarea de Ingeniería. Registrar matrimonio.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 4	<b>Numero de historia de usuario:</b> 2
<b>Nombre tarea:</b> Registrar matrimonio	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 06/05/2019	<b>Fecha fin:</b> 06/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “registrar matrimonio”.</p> <p>Ingresa los atributos: nombre de la parroquia, fecha del matrimonio, nombre del párroco, nombre del contrayente, nombre de la contrayente, nombre del padre y madre del contrayente, nombre del padre y madre de la contrayente, nombre de los padrinos de cada matrimonio.</p> <p>Hace clic en el botón “registrar”.</p> <p>Los atributos ingresados se almacenan en la base de datos.</p> <p>La aplicación web muestra el mensaje “grabación correcta”.</p> <p>Observaciones: Cuando no se graba correctamente se muestra el mensaje “Error de registro”.</p>	

Tabla 4.19. Tarea de Ingeniería. Modificar matrimonio.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 5	<b>Numero de historia de usuario:</b> 2
<b>Nombre tarea:</b> Modificar matrimonio	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 07/05/2019	<b>Fecha fin:</b> 07/05/2019
<b>Programador responsable:</b> Programador – Asistente	

<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “modificar matrimonio”, para ingresar el atributo clave del matrimonio.</p> <p>Hace clic en el botón “buscar” y los atributos: nombre de la parroquia, fecha del matrimonio, nombre del párroco, nombre del contrayente, nombre de la contrayente, nombre del padre y madre del contrayente, nombre del padre y madre de la contrayente, nombre de los padrinos de cada matrimonio, se muestran en la interfaz “modificar matrimonio”.</p> <p>Modifica algunos atributos hace clic en el botón “modificar”.</p> <p>Los atributos modificados se almacenan en la base de datos.</p> <p>La aplicación web muestra el mensaje “modificación correcta”.</p> <p>Observaciones: Cuando no se modifica correctamente se muestra el mensaje “Error de modificación”.</p>
---

Tabla 4.20. Tarea de Ingeniería. Eliminar matrimonio.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 6	<b>Numero de historia de usuario:</b> 2
<b>Nombre tarea:</b> Eliminar matrimonio	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 08/05/2019	<b>Fecha fin:</b> 08/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “eliminar matrimonio”, para ingresar el atributo clave de un matrimonio.</p> <p>Hace clic en el botón “buscar”, los atributos: nombre de la parroquia, fecha del matrimonio, nombre del párroco, nombre del contrayente, nombre de la contrayente, nombre del padre y madre del contrayente, nombre del padre y madre de la contrayente, nombre de los padrinos de cada matrimonio se muestran en la interfaz “eliminar matrimonio”.</p> <p>El archivero hace clic en el botón “eliminar”.</p> <p>El bautismo se elimina de la base de datos.</p> <p>La aplicación web muestra el mensaje “eliminación correcta”.</p> <p>Observaciones: Si no se elimina se muestra el mensaje “Error de eliminación”.</p>	

Tabla 4.21. Tarea de Ingeniería. Registrar defunción.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 7	<b>Numero de historia de usuario:</b> 3
<b>Nombre tarea:</b> Registrar matrimonio	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 09/05/2019	<b>Fecha fin:</b> 09/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “registrar defunción”.</p> <p>Ingresa los atributos: nombre y apellidos, fecha de defunción, edad, nombre de los padres del difunto, lugar de procedencia del difunto, causa del fallecimiento para cada defunción.</p> <p>Hace clic en el botón “registrar”.</p> <p>Los atributos ingresados se almacenan en la base de datos.</p> <p>La aplicación web muestra el mensaje “grabación correcta”.</p>	
<p>Observaciones: Cuando no se graba correctamente se muestra el mensaje “Error de registro”.</p>	

Tabla 4.22. Tarea de Ingeniería. Modificar defunción.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 8	<b>Numero de historia de usuario:</b> 3
<b>Nombre tarea:</b> Modificar matrimonio	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 10/05/2019	<b>Fecha fin:</b> 10/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “modificar defunción”, para ingresar el atributo clave de defunción.</p> <p>Hace clic en el botón “buscar” y los atributos: nombre y apellidos, fecha de defunción, edad, nombre de los padres del difunto, lugar de procedencia del difunto, causa del fallecimiento, se muestran en la interfaz “modificar defunción”.</p> <p>Modifica algunos atributos, hace clic en el botón “modificar”.</p>	

Los atributos modificados se almacenan en la base de datos.
La aplicación web muestra el mensaje “modificación correcta”.
Observaciones: Cuando no se modifica correctamente se muestra el mensaje “Error de modificación”.

Tabla 4.23. Tarea de Ingeniería. Eliminar defunción.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 9	<b>Numero de historia de usuario:</b> 3
<b>Nombre tarea:</b> Eliminar defunción	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 11/05/2019	<b>Fecha fin:</b> 11/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “eliminar defunción”, para ingresar el atributo clave de una defunción.</p> <p>Hace clic en el botón “buscar”, los atributos nombre y apellidos, fecha de defunción, edad, nombre de los padres del difunto, lugar de procedencia del difunto, causa del fallecimiento, se muestran en la interfaz “eliminar defunción”.</p> <p>El archivero hace clic en el botón “eliminar”.</p> <p>El bautismo se elimina de la base de datos.</p> <p>La aplicación web muestra el mensaje “eliminación correcta”.</p>	
Observaciones: Cuando no se elimina correctamente se muestra el mensaje “Error de eliminación”.	

Tabla 4.24. Tarea de Ingeniería. Registrar confirmación.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 10	<b>Numero de historia de usuario:</b> 4
<b>Nombre tarea:</b> Registrar confirmación	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 12/05/2019	<b>Fecha fin:</b> 12/05/2019
<b>Programador responsable:</b> Programador – Asistente	
Descripción:	

<p>El archivero usa la interfaz “registrar confirmación”.</p> <p>Ingresar los atributos: fecha de bautismo, lugar del bautismo, nombre del confirmado, nombre de los padres, nombre de los padrinos, ministerio de la confirmación y fecha de cada confirmación.</p> <p>Hace clic en el botón “registrar”.</p> <p>Los atributos ingresados se almacenan en la base de datos.</p> <p>La aplicación web muestra el mensaje “grabación correcta”.</p>
<p>Observaciones: Cuando no se graba correctamente se muestra el mensaje “Error de registro”.</p>

Tabla 4.25. Tarea de Ingeniería. Modificar confirmación.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 11	<b>Numero de historia de usuario:</b> 4
<b>Nombre tarea:</b> Modificar confirmación	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 13/05/2019	<b>Fecha fin:</b> 13/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “modificar confirmación”, para ingresar el atributo clave de confirmación.</p> <p>Hace clic en el botón “buscar” y los atributos número de confirmado, fecha de bautismo, lugar del bautismo, nombre del confirmado, nombre de los padres, nombre de los padrinos, ministerio de la confirmación y fecha, se muestran en la interfaz “modificar confirmación”.</p> <p>Modifica algunos atributos, hace clic en el botón “modificar”.</p> <p>Los atributos modificados se almacenan en la base de datos.</p> <p>La aplicación web muestra el mensaje “modificación correcta”.</p>	
<p>Observaciones: Cuando no se modifica correctamente se muestra el mensaje “Error de modificación”.</p>	

Tabla 4.26. Tarea de Ingeniería. Eliminar confirmación.

<b>TAREA DE INGENIERÍA</b>
----------------------------

<b>Numero de tarea:</b> 12	<b>Numero de historia de usuario:</b> 4
<b>Nombre tarea:</b> Eliminar confirmación	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 14/05/2019	<b>Fecha fin:</b> 14/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “eliminar confirmación”, para ingresar el atributo clave de una confirmación.</p> <p>Hace clic en el botón “buscar”, los atributos número de confirmado, fecha de bautismo, lugar del bautismo, nombre del confirmado, nombre de los padres, nombre de los padrinos, ministerio de la confirmación y fecha, se muestran en la interfaz “eliminar confirmación”.</p> <p>El archivero hace clic en el botón “eliminar”.</p> <p>La confirmación se elimina de la base de datos.</p> <p>La aplicación web muestra el mensaje “eliminación correcta”.</p>	
Observaciones: Cuando no se elimina correctamente se muestra el mensaje “Error de eliminación”.	

Tabla 4.27. Tarea de Ingeniería. Almacenar imagen.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 13	<b>Numero de historia de usuario:</b> 5
<b>Nombre tarea:</b> Almacenar imagen	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 15/05/2019	<b>Fecha fin:</b> 16/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero usa la interfaz “almacenar imagen”.</p> <p>Selecciona del botón de combo, el tipo de fondo documental, e ingresa el atributo clave del tipo de fondo.</p> <p>Hace clic en el botón “cargar imagen”, busca la imagen.</p> <p>Hace clic en el botón “registrar imagen”.</p> <p>La imagen se almacena en la base de datos.</p> <p>La aplicación web muestra el mensaje “registro correcto”.</p>	

Observaciones: Cuando no se graba correctamente se muestra el mensaje “Error de registro”.
--

Tabla 4.28. Tarea de Ingeniería. Consultar fondo documental.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 14	<b>Numero de historia de usuario:</b> 6, 7, 8
<b>Nombre tarea:</b> Consultar fondo documental	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 8
<b>Fecha inicio:</b> 17/05/2019	<b>Fecha fin:</b> 24/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<b>Descripción:</b> El archivero usa la interfaz “consultar fondo” y selecciona el tipo de fondo. El archivero ingresa el texto de búsqueda del fondo y hace clic en botón “buscar”. Se muestra los atributos según tipo de fondo.	
Observaciones: Cuando no se encuentra un fondo se muestra el mensaje “No existe fondo”.	

Tabla 4.29. Tarea de Ingeniería. Consultar imagen.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 15	<b>Numero de historia de usuario:</b> 10
<b>Nombre tarea:</b> Consultar imagen	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 25/05/2019	<b>Fecha fin:</b> 26/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<b>Descripción:</b> El archivero usa la interfaz “consultar imagen” y selecciona el tipo de fondo. El archivero ingresa texto de búsqueda del fondo y hace clic en botón “buscar”. Se muestra la imagen según el tipo de fondo.	
Observaciones: Cuando no se encuentra un fondo se muestra el mensaje “No existe fondo”.	

imagen”.
----------

Tabla 4.30. Tarea de Ingeniería. Iniciar sesión.

<b>TAREA DE INGENIERÍA</b>	
<b>Numero de tarea:</b> 16	<b>Numero de historia de usuario:</b> 11
<b>Nombre tarea:</b> Iniciar sesión	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 27/05/2019	<b>Fecha fin:</b> 28/05/2019
<b>Programador responsable:</b> Programador – Asistente	
<p><b>Descripción:</b></p> <p>El archivero o ciudadano usa la interfaz “Iniciar sesión”.</p> <p>El archivero o ciudadano ingresa su usuario y clave, hace clic en botón “ingresar”.</p> <p>El sistema muestra el menú del fondo documental.</p>	
<p>Observaciones: Cuando se equivoca el usuario o clave, el sistema muestra el mensaje “Error de clave”.</p>	

### PLAN DE ITERACIÓN (Primera)

Tabla 4.31. Plan de Iteración (Primera)

Nº	HISTORIA DE USUARIO	TAREA DE INGENIERÍA
1	Mantener datos de cada bautismo.	Registrar bautismo.
1	Mantener datos de cada bautismo.	Modificar bautismo.
1	Mantener datos de cada bautismo.	Eliminar bautismo.
2	Mantener datos de cada matrimonio.	Registrar matrimonio.
2	Mantener datos de cada matrimonio.	Modificar matrimonio.
2	Mantener datos de cada matrimonio.	Eliminar matrimonio.
3	Mantener datos de cada defunción.	Registrar defunción.
3	Mantener datos de cada defunción.	Modificar defunción.
3	Mantener datos de cada defunción.	Eliminar defunción.
4	Mantener datos de cada confirmación.	Registrar confirmación.



4	Mantener datos de cada confirmación.	Modificar confirmación.
4	Mantener datos de cada confirmación.	Eliminar confirmación.
5	Almacenar la imagen de un fondo documental.	Almacenar imagen.

### PLAN DE ITERACIÓN (Segunda)

Tabla 4.32. Plan de Iteración (Segunda)

Nº	HISTORIA DE USUARIO	TAREA DE INGENIERÍA
6, 7, 8 y 9	Consultar los bautismos, matrimonios, defunciones y confirmaciones.	Consultar fondo.
10	Consultar la imagen de un fondo documental.	Consultar imagen.
11	Iniciar sesión de usuarios a la aplicación	Iniciar sesión.

Tabla N° 4.33. Plan de iteración clasificada por fechas de desarrollo

TAREA DE INGENIERÍA	FECHA INICIO	FECHA FIN	RESPONSABLE
1	02/05/2019	02/05/2019	Programador - Asistente
2	03/05/2019	03/05/2019	Programador - Asistente
3	04/05/2019	04/05/2019	Programador - Asistente
4	05/05/2019	05/05/2019	Programador - Asistente
5	06/05/2019	06/05/2019	Programador - Asistente
6	07/05/2019	07/05/2019	Programador - Asistente
7	08/05/2019	08/05/2019	Programador - Asistente
8	09/05/2019	09/05/2019	Programador - Asistente
9	10/05/2019	10/05/2019	Programador - Asistente
10	11/05/2019	11/05/2019	Programador - Asistente

11	12/05/2019	12/05/2019	Programador - Asistente
12	13/05/2019	13/05/2019	Programador - Asistente
13	14/05/2019	21/05/2019	Programador - Asistente
14	22/05/2019	23/05/2019	Programador - Asistente
15	24/05/2019	25/05/2019	Programador - Asistente
16	26/05/2019	27/05/2019	Programador - Asistente

## INTERFAZ DE USUARIO

Figura 4.6. Interfaz Iniciar Sesión de la aplicación web

Figura 4.7. Interfaz Agregar Usuario



Figura 4.8. Interfaz buscar Fondo Documental

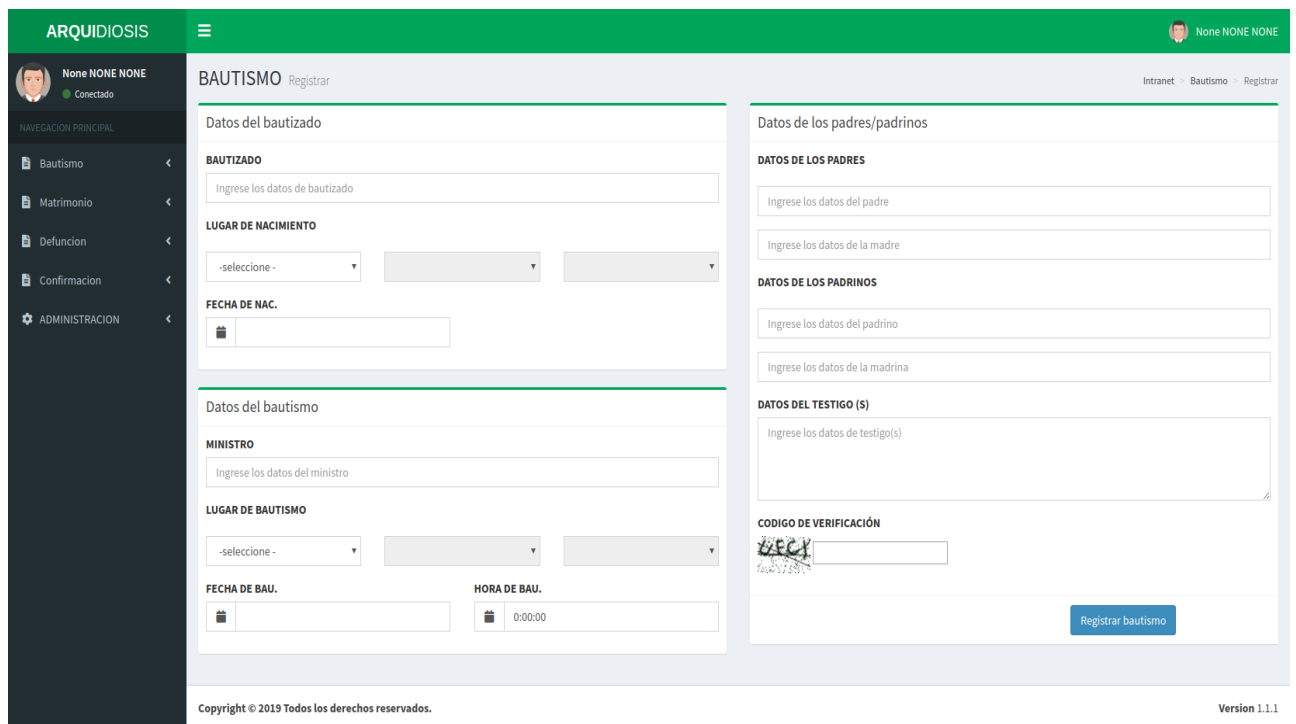


Figura 4.9. Interfaz Registrar Bautismo

ARQUIDIOSIS None NONE NONE

None NONE NONE Intranet > Matrimonio > Registrar

Conectado

NAVEGACION PRINCIPAL

- Bautismo <
- Matrimonio <
- Defuncion <
- Confirmacion <

ADMINISTRACION <

### MATRIMONIO Registrar

**Datos de los contrayentes**

**ESPOSO**

**ESPOSA**

**Datos de los padres/padrinos**

**DATOS DE LOS PADRES**

**Datos del matrimonio**

**PARROQUIA**

**PARROCO**

**LUGAR DE MATRIMONIO**

-seleccione-

**FECHA DE MAT.**  **HORA DE MAT.**

**DATOS DE LOS PADRINOS**

Figura 4.10. Interfaz Registrar Matrimonio

ARQUIDIOSIS None NONE NONE

None NONE NONE Intranet > Defuncion > Registrar

Conectado

NAVEGACION PRINCIPAL

- Bautismo <
- Matrimonio <
- Defuncion <
- Confirmacion <

ADMINISTRACION <

### DEFUNCION Registrar

**Datos del difunto**

**DIFUNTO**

**EDAD**

**Datos de los padres**

**DATOS DE LOS PADRES**

**Datos de la defuncion**

**CAUSA**

**LUGAR DE DEFUNCIÓN**

-seleccione-

**FECHA DE DEF.**  **HORA DE DEF.**

**CODIGO DE VERIFICACIÓN**

XCEP

[Registrar defuncion](#)

Figura 4.11. Interfaz Registrar Defunción

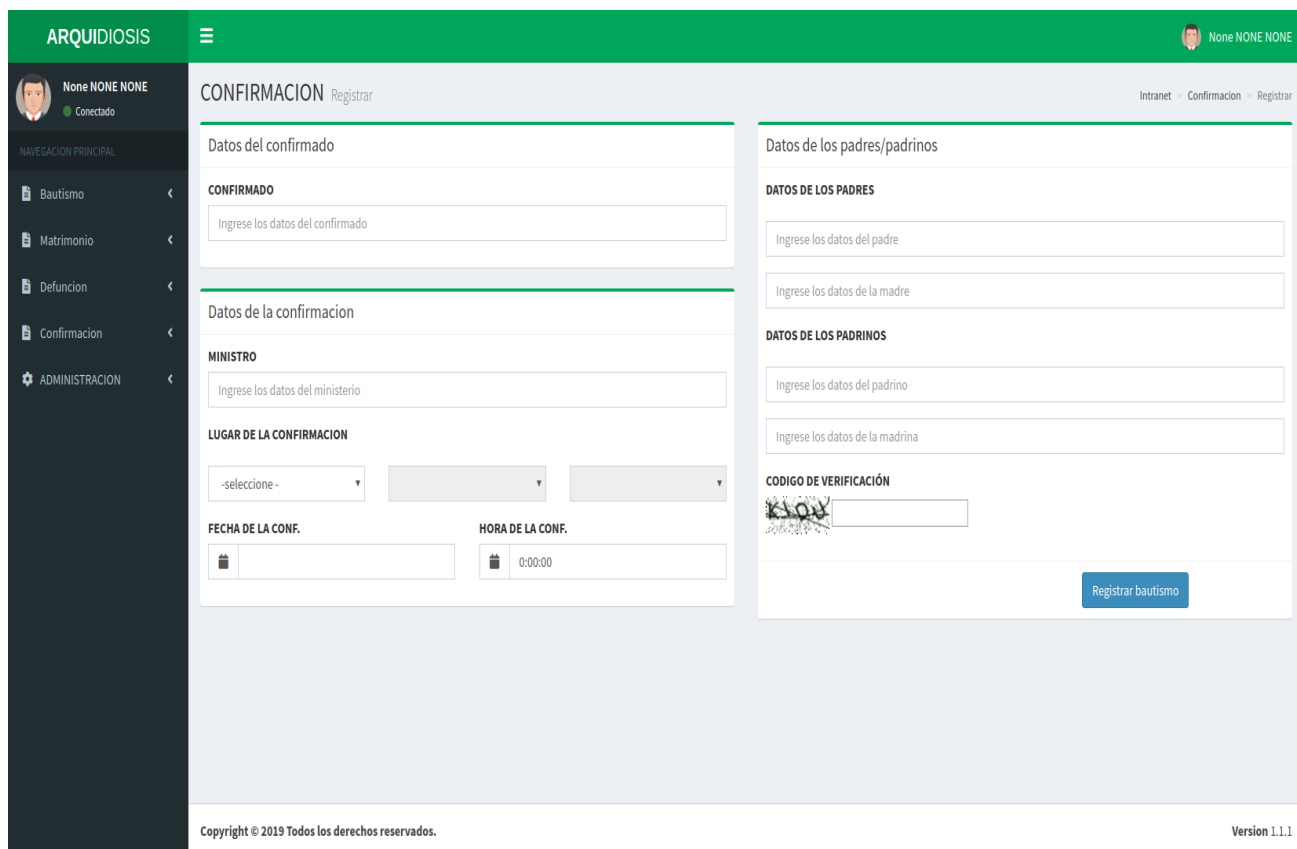


Figura 4.12. Interfaz Registrar Confirmación

## TARJETAS CLASE RESPONSABILIDAD COLABORACIÓN (CRC)

Tabla 4.34. Clase Archivero

CLASE: Archivero	
<p><b>Responsabilidades:</b> Mantener datos del archivero.</p>	<p><b>Colaboradores:</b> Bautismo Matrimonio Defuncion Confirmacion Fondo_Documental</p> <p><b>Atributos:</b> ANIO CANTTRXCARGA ESTADO FECCRE FECFIN</p>

Tabla 4.35. Clase Bautismo

<b>CLASE: Bautismo</b>	
<p><b>Responsabilidades:</b> Mantener datos de cada bautismo.</p>	<p><b>Colaboradores:</b> Archivero Ciudadano</p> <p><b>Atributos:</b> ANIO MES ESTADO FECLECTU IDCABMEDI</p>

Tabla 4.36. Clase Matrimonio

<b>CLASE: Matrimonio</b>	
<p><b>Responsabilidades:</b> Mantener datos de cada matrimonio.</p>	<p><b>Colaboradores:</b> Archivero Ciudadano</p> <p><b>Atributos:</b> ANIO MES ESTADO FECLECTU IDCABMEDI</p>

Tabla 4.37. Clase Defunción

<b>CLASE: Defunción</b>	
<p><b>Responsabilidades:</b> Mantener datos de cada defunción.</p>	<p><b>Colaboradores:</b> Archivero Ciudadano</p> <p><b>Atributos:</b> ANIO MES</p>

	ESTADO FECLECTU IDCABMEDI
--	---------------------------------

Tabla 4.38. Clase Confirmación

<b>CLASE: Confirmación</b>	
<p><b>Responsabilidades:</b> Mantener datos de cada confirmación.</p>	<p><b>Colaboradores:</b> Archivero Ciudadano</p> <p><b>Atributos:</b> ANIO MES ESTADO FECLECTU IDCABMEDI</p>

Tabla 4.39. Clase Fondo\_Documental

<b>CLASE: Fondo_Documental</b>	
<p><b>Responsabilidades:</b> Mantener datos del fondo documental.</p>	<p><b>Colaboradores:</b> Bautismo Matrimonio Defuncion Confirmacion Archivero</p> <p><b>Atributos:</b> ANIO MES ESTADO FECLECTU IDCABMEDI</p>

Tabla 4.40. Clase Ciudadano

<b>CLASE: Ciudadano</b>	
<p><b>Responsabilidades:</b> Mantener datos del ciudadano.</p>	<p><b>Colaboradores:</b> Bautismo Matrimonio Defuncion Confirmacion</p> <p><b>Atributos:</b> ANIO MES ESTADO FECLECTU IDCABMEDI</p>





## CÓDIGO FUENTE

```
from django.db import models
from Usuario.models import User, Ubigeo
from django.db import transaction
from django.db.models.signals import post_save
from django.dispatch import receiver
# Create your models here.
# fondos documentales
class Bautismo(models.Model):
    codigo = models.CharField(max_length=20)
    bautizado = models.CharField(max_length=150)
    ministro = models.CharField(max_length=150)
    padre = models.TextField()
    padrino = models.TextField()
    testigo = models.TextField()
    lugar_bautismo = models.ForeignKey(Ubigeo, on_delete=models.PROTECT,
related_name='bautismolugarbautismo')
    lugar_nacimiento = models.ForeignKey(Ubigeo, on_delete=models.PROTECT)
    fecha_bautismo = models.DateField()
    hora_bautismo = models.TimeField()
    fecha_nacimiento = models.DateField()
    imagen = models.ImageField(upload_to='ImagenFondoDocumental/', blank=True, null=True)
    usuario = models.ForeignKey(User, on_delete=models.PROTECT)
    fecha_registro = models.DateTimeField(auto_now_add=True)
    fecha_actualizacion = models.DateTimeField(auto_now=True)
    flag = models.BooleanField(default=True)
    def __str__(self):
        return self.codigo
    def get_padres(self):
        return self.padre.split(';')
    def get_padres_initial(self):
        padre = self.padre.replace('(PADRE)', '')
        padre = padre.replace('(MADRE)', '')
        return padre.split(';')
    def get_padrinos(self):
        return self.padrino.split(';')
    def get_padrinos_initial(self):
        padrino = self.padrino.replace('(PADRINO)', '')
        padrino = padrino.replace('(MADRINA)', '')
```

```

        return padrino.split(';')

    def get_fecha_nacimiento_init(self):
        return '{}{/}/{}'.format(self.fecha_nacimiento.day, self.fecha_nacimiento.month,
self.fecha_nacimiento.year)

    def get_fecha_bautismo_init(self):
        return '{}{/}/{}'.format(self.fecha_bautismo.day, self.fecha_bautismo.month,
self.fecha_bautismo.year)

    class Meta:
        ordering = ['-id']

    @receiver(post_save, sender=Bautismo)
    def create_code_bau(sender, instance, created, **kwargs):
        if created:
            instance.codigo = 'B{}'.format(str(instance.id).zfill(5), str(instance.fecha_bautismo.year))
            instance.save()

class Matrimonio(models.Model):
    codigo = models.CharField(max_length=20)
    parroquia = models.CharField(max_length=200)
    parroco = models.CharField(max_length=200)
    contrayente = models.TextField()
    padre = models.TextField()
    padrino = models.TextField()
    lugar_matrimonio = models.ForeignKey(Ubigeo, on_delete=models.PROTECT)
    fecha_matrimonio = models.DateField()
    hora_matrimonio = models.TimeField()
    imagen = models.ImageField(upload_to=ImagenFondoDocumental/, blank=True, null=True)
    usuario = models.ForeignKey(User, on_delete=models.PROTECT)
    fecha_registro = models.DateTimeField(auto_now_add=True)
    fecha_actualizacion = models.DateTimeField(auto_now=True)
    flag = models.BooleanField(default=True)

    def __str__(self):
        return self.codigo

    def get_padres(self):
        return self.padre.split(';')

    def get_padres_initial(self):
        padre = self.padre.replace('(PADRE)', '')
        padre = padre.replace('(MADRE)', '')
        return padre.split(';')

    def get_padrinos(self):
        return self.padrino.split(';')

    def get_padrinos_initial(self):

```

```

        padrino = self.padrino.replace('(PADRINO)', '')
        padrino = padrino.replace('(MADRINA)', '')
        return padrino.split(';')
def get_contrayente(self):
    return self.contrayente.split(';')
def get_contrayente_initial(self):
    padrino = self.contrayente.replace('(ESPOSO)', '')
    padrino = padrino.replace('(ESPOSA)', '')
    return padrino.split(';')
def get_fecha_matrimonio_init(self):
    return '{}{/}/{}'.format(self.fecha_matrimonio.day, self.fecha_matrimonio.month,
self.fecha_matrimonio.year)
class Meta:
    ordering = ['-id']
@receiver(post_save, sender=Matrimonio)
def create_code_matri(sender, instance, created, **kwargs):
    if created:
        instance.codigo = 'M{ }{}'.format(str(instance.id).zfill(5),
str(instance.fecha_matrimonio.year))
        instance.save()
class Defuncion(models.Model):
    codigo = models.CharField(max_length=20)
    difunto = models.CharField(max_length=150)
    edad = models.PositiveSmallIntegerField()
    padre = models.TextField()
    causa = models.CharField(max_length=100)
    lugar_defuncion = models.ForeignKey(Ubigeo, on_delete=models.PROTECT)
    fecha_defuncion = models.DateField()
    hora_defuncion = models.TimeField()
    imagen = models.ImageField(upload_to='ImagenFondoDocumental/', blank=True, null=True)
    usuario = models.ForeignKey(User, on_delete=models.PROTECT)
    fecha_registro = models.DateTimeField(auto_now_add=True)
    fecha_actualizacion = models.DateTimeField(auto_now=True)
    flag = models.BooleanField(default=True)
def __str__(self):
    return self.codigo
def get_padres(self):
    return self.padre.split(';')
def get_padres_initial(self):
    padre = self.padre.replace('(PADRE)', '')

```

```

        padre = padre.replace('(MADRE)', '')
        return padre.split(';')

    def get_fecha_defuncion_init(self):
        return '{}{/}/{}'.format(self.fecha_defuncion.day, self.fecha_defuncion.month,
self.fecha_defuncion.year)

    class Meta:
        ordering = ['-id']

@receiver(post_save, sender=Defuncion)
def create_code_def(sender, instance, created, **kwargs):
    if created:
        instance.codigo = 'D{}'.format(str(instance.id).zfill(5), str(instance.fecha_defuncion.year))
        instance.save()

class Confirmacion(models.Model):
    codigo = models.CharField(max_length=20)
    confirmado = models.CharField(max_length=150)
    padre = models.TextField()
    padrino = models.TextField()
    ministerio = models.CharField(max_length=200)
    lugar_confirmacion = models.ForeignKey(Ubigeo, on_delete=models.PROTECT)
    fecha_confirmacion = models.DateField()
    hora_confirmacion = models.TimeField()
    imagen = models.ImageField(upload_to='ImagenFondoDocumental/', blank=True, null=True)
    usuario = models.ForeignKey(User, on_delete=models.PROTECT)
    fecha_registro = models.DateTimeField(auto_now_add=True)
    fecha_actualizacion = models.DateTimeField(auto_now=True)
    flag = models.BooleanField(default=True)

    def __str__(self):
        return self.codigo

    def get_padres(self):
        return self.padre.split(';')

    def get_padres_initial(self):
        padre = self.padre.replace('(PADRE)', '')
        padre = padre.replace('(MADRE)', '')
        return padre.split(';')

    def get_padrinos(self):
        return self.padrino.split(';')

    def get_padrinos_initial(self):
        padrino = self.padrino.replace('(PADRINO)', '')
        padrino = padrino.replace('(MADRINA)', '')
        return padrino.split(';')

```

```

def get_fecha_confirmacion_init(self):
    return '{}/{}/{}'.format(self.fecha_confirmacion.day, self.fecha_confirmacion.month,
                              self.fecha_confirmacion.year)

class Meta:
    ordering = ['-id']

@receiver(post_save, sender=Confirmacion)
def create_code_conf(sender, instance, created, **kwargs):
    if created:
        instance.codigo = 'C{}'.format(str(instance.id).zfill(5),
str(instance.fecha_confirmacion.year))
        instance.save()

```

Figura 4.14. Código Fuente de la aplicación web y móvil

## PRUEBAS DE ACEPTACIÓN

Tabla 4.41. Reporte de prueba de aceptación. Iniciar Sesión de usuarios de la aplicación

<b>Nº CASO PRUEBA</b>	1
Propósito	Iniciar sesión de usuarios de la aplicación web y movil
<b>ACTIVIDAD</b>	
Inicialización	En el navegador ingresar la ruta de inicio de sistema
Descripción de datos de entrada	Nombre de usuario, clave.
<b>RESULTADOS</b>	
Esperado	Sesión de usuario inicializada correctamente
Reales	El usuario inicia sesión y abre las opciones del sistema

Tabla 4.42. Reporte de pruebas de aceptación. Consultar fondo documental

<b>Nº CASO PRUEBA</b>	2
Propósito	Comprobar que la aplicación web y móvil muestra los datos de los involucrados en los registros de bautismo, matrimonio, defunción y confirmación.
<b>ACTIVIDAD</b>	
Inicialización	Seleccionar la opción ingresar bautismo o matrimonio, o defunción o confirmación; en el menú de la aplicación web.
Descripción de datos de entrada	Datos de los involucrados como: nombres y apellidos, edad, de los involucrados en el bautismo, matrimonio, defunción y

	confirmación.
<b>RESULTADOS</b>	
Esperado	Muestra todos los datos ingresados.
Reales	Muestra los datos realmente instanciados.

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **5.1. CONCLUSIONES**

- a) Se ha logrado automatizar el fondo documental bautismo, mediante las historias de usuario 1, 6 y 10, interfaz gráfica figura 4.8 y 4.9, la base de datos y el código fuente correspondiente, que permite registrar, modificar y consultar los bautizos.
- b) Se ha logrado automatizar el fondo documental matrimonio, mediante las historias de usuario 2, 7 y 10 interfaz gráfica figura 4.8 y 10, la base de datos y el código fuente correspondiente, que permite registrar, modificar y consultar los bautizos.
- c) Se ha logrado automatizar el fondo documental defunciones, mediante las historias de usuario 3, 8 y 10, interfaz gráfica figura 4.8 y 4.11, la base de datos y el código fuente correspondiente, que permite registrar, modificar y consultar los bautizos.
- d) Se ha logrado automatizar el fondo documental confirmación, mediante las historias de usuario 4, 9 y 10, interfaz gráfica figura 4.8 y 4.12, la base de datos y el código fuente correspondiente, que permite registrar, modificar y consultar los bautizos.

#### **5.2. RECOMENDACIONES**

- a) Se recomienda desarrollar una aplicación web para los fondos documentales desde el año 1540 hasta 1900, que considere todos los fondos y la digitalización de alta resolución de los documentos a fin de evitar su deterioro y tener consultas en línea.
- b) Se debe estudiar el desarrollo de una aplicación móvil para las consultas de todos los fondos documentales y el acceso a estos mediante el pago en línea, firmas digitales e impresión.



## BIBLIOGRAFÍA

1. Academia Española, (2001). *Diccionario de la Real Academia de la Lengua Española*. (21ª Ed.). Madrid: Real Academia Española.
2. Acón, A., Trujillo, A., Guido, H. (2011). Implementación de un servicio web en la UNED, herramienta para lograr excelencia académica. *Revista de Calidad en la Educación Superior*. 2(2), 193-211.
3. Aumaille, B. (2002). *J2EE Desarrollo de aplicaciones web*. Barcelona, España: Eni.
4. Archivo General de la Nación, (s.f.) *Normas técnicas del sistema nacional de archivos*. Lima, Perú.
5. Archivo General de la Nación. (2008). Reglamento de aplicaciones administrativas por infracciones en contra del patrimonio documental archivístico y cultural de la Nación. Lima, Perú.
6. Arellano, S. (2015). *Matrimonio*. Recuperado el 14 de octubre de 2017, de <https://archivos.juridicas.unam.mx/www/bjv/libros/8/3834/13.pdf>
7. Archivo Nacional de Chile. (2016, abril). Recuperado el 28 de octubre de 2017 de <http://www.archivonacional.cl/616/w3-propertyvalue-47701.html>.
8. Bavaresco, A. (2006). *Proceso Metodológico en la Investigación. (Cómo hacer un diseño de investigación)*. Maracaibo: La Universidad del Zulia.
9. Batini, C., Ceri, S. y Navathe, S. B. (1994). *Diseño conceptual de base de datos. Un enfoque de entidades-interrelaciones*. Boston, Estados Unidos: Addison-Wesley Iboamericana.
10. Beck, K. (1999). *Extreme Programming Explained*. Addison- Wesley.
11. Beck, K., & Fowler, M. (2002). *Planning Extreme Programming*. Addison Wesley.
12. Borrás J. (2001). Las relaciones entre archiveros y productores de documentos. España: Archivo de la Universidad Pompeu Fabra.
13. Campos, S. (2006). *Aprenda ya. Programación orientada a objetos*. Lima, Perú: Editorial Universitaria.
14. Ceballos, F. (1993). *Programación orientada a objetos con C++*. Washington, Estados Unidos: Addison - Wesley Iberoamericana.
15. Comer, D. (1995). *El libro de internet. Todo lo que usted desea saber sobre*

*redes de computadoras y acerca de cómo funciona internet.* México, D.F., México: Prentice Hall Hispanoamericana.

16. Cortés, C. (1982). *El Archivo Histórico del Atlántico. El Caribe: Escuela de La Danza Folclórica.*
17. Diócesis de Victoria. (s.f.) *Archivo Histórico Diocesano de Victoria.* Recuperado el 14 de octubre de 2017, de <http://www.ahdv-geah.org/cast/glosario.html>.
18. Domínguez, A. (2010). *El Archivo Arzobispal.* Manuscrito no publicado, Arzobispado de Ayacucho, Ayacucho, Perú.
19. Fernández, G. (2002). *Introducción a Extreme Programming, Ingeniería de Software II.* Madrid, España.
20. Galende, D. (2011). *Los Libros Sacramentales de Defunción en el Madrid Moderno.* Recuperado el 14 de octubre de 2017. de <http://pendientedemigracion.ucm.es/centros/cont/descargas/documento25275.pdf>
21. García, M. (2007). *Apuntes sobre los Archivos parroquiales en España.* Recuperado el 14 de octubre de 2017, de <http://sisbib.unmsm.edu.pe/bibvirtualdata/publicaciones/biblios/n34/a07n34.pdf>
22. Garcerant, I. (2008). *Modelo de Dominio.* Recuperado el 19 de noviembre de 2017, de <https://synergix.wordpress.com/2008/07/10/modelo-de-dominio/>
23. Garro, J. (2007) *El Proyecto de Digitalización y Difusión de los Archivos Históricos Diocesano: debate y perspectivas de futuro.* Bilbao, 8 y 9 de noviembre.
24. González, M. y Navarro, C. (2010, Agosto). portal de archivos españoles, Generando puentes entre el investigador y los fondos archivísticos. *Investigación bibliotecológica*, 51, artículo 24. Recuperado el 28 de octubre de 2017, de [http://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S0187-358X2010000200003](http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2010000200003)
25. Gómez, J. (s.f.). *Series documentales.* Archivo Diocesano de Tuy, España.
26. Gutiérrez, J., Escalona, J., Mejías, M., Torres, J. (2004). *Pruebas del sistema en programación extrema.* Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla. España.
27. Greg, V. (1994). *Programación Orientada a Objetos una Introducción.* México, D.F., México: MacGraw – Hill/ Interamericana de

México.

28. Hansler, G. y Hansen, J. (1997). *Diseño y administración de base de datos* (2ª Ed.). Madrid, España: Prentice Hall.
29. Iglesia en España. (2001). Vol. I Barcelona 2001.
30. Jeffries, R., Anderson, K., Hendrickson, Chet (2000). *Extreme Programming Installed*. Boston, USA: Addison-Wesley.
31. Joyanes, L. (1998). *Programación orientada a objetos* (2ª Ed.). España: Mcgraw W-Hill/Interamericana De España.
32. Juárez, R. (2011). *Proceso de desarrollo independiente de una aplicación móvil android*. Tesis de pregrado. Universidad Nacional Autónoma de México, México, México.
33. Laudon, K., & Laudon, J. P. (2004). *Sistemas de información gerencial: administración de la empresa digital*. México: Pearson Educación.
34. Letelier, P., Penadés, C. (2006). *Métodologías ágiles para el desarrollo de software: Extreme Programming XP*. Universidad Politécnica de Valencia. Valencia. España.
35. Lucas, A., Romera, P., Fraile, M., Argente, F. y Alfaro, A. (1993). *Diseño y Gestión de Sistema de Base de Datos*. Madrid, España: Editorial Paraninfo.
36. Luján, S. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web*. Madrid, España: Club Universitario.
37. Luján, S. (2001). *Programación en Internet: cliente Web*. Madrid, España: Club universitario.
38. Luque, I., Gómez, M., López, N. y Cerruela, G. (2002). *Base de Datos. Desde Chen hasta Codd con Oracle. ORACLE*. México, D.F., México: Alfaomega Grupo Editor.
39. Mukhar, K., Lavinger, T. y Carnell, J. (2002). *Fundamentos de base de datos con Java*. Madrid, España: Ediciones Anaya Multimedia.
40. Naughton, P. (1996). *Manual de Java*. Madrid, España: McGraw – Hill/Interamericana de España.
41. Noriega, A. (2007). *Programación en Java2*. Lima, Perú: Grupo Editorial Megabyte.
42. Porras Efraín (2010). Comparación de dos Procesos de Desarrollo de Software usando Los Métodos Iconix y XP, Caso: Comercialización de la Tara en La Región Ayacucho. Tesis de Maestría. Universidad Nacional de Ingeniería,

Lima, Perú.

43. Quiroz, D. (2013). *Desarrollo de aplicación móvil para verificar el estado de las colas de los usuarios, para una entidad financiera*. Tesis de pregrado. Universidad José Antonio Páez, San Diego, Venezuela.
44. Romero, N. (1994). *El archivo histórico*. México: McGraw-Hill Interamericana
45. Sánchez, S. (2011). *Metodologías de Análisis y Diseño*. Recuperado el 19 de noviembre de 2017, de <https://es.slideshare.net/SergioRios/unidad-5-mad-modelado-analisis-modelo-conceptual>
46. Tamayo, M. y Tamayo, A. (1997). *El Proceso de la Investigación Científica. (Tercera Edición)*. México: LIMUSA.
47. Téllez, N. (2014). la nueva gestión de archivos. *Gestión documental y software libre en el archivo Histórico del centro excursionista de Cataluña (ahcec): la descripción de sus fondos Documentales a través de la implantación del Programa ica-atom*. Recuperado el 28 de octubre de 2017, de [https://archivosierrapambley.files.wordpress.com/2015/02/nuria\\_tellez\\_2014.pdf](https://archivosierrapambley.files.wordpress.com/2015/02/nuria_tellez_2014.pdf).
48. Vásquez, J. y Balta, V. (2008). *Super JAVA Estándar Edition for Windows with JCreator IDE (3ª Ed.)*. Lima, Perú: La Esperanza. Diseños e Impresión SAC.
49. Vargas, R. y Maltés, J. (s.f.). *Programación en capas*. San José, Costa Rica: Publicaciones Vértice.
50. Vargas Del Valle, R., & Maltés Granados, J. (2007). *Programación en Capas*. San José: Universidad de Costa Rica, Ciencias de Computación e Informática.
51. Stallings, W. (2006). *Comunicaciones y Redes de Computadores*. Sexta Edición.

## ANEXOS

### ANEXO 1. MATRIZ DE OPERACIONALIZACIÓN DE VARIABLES

Variables	Variable de Interés	Indicadores	Ítems	Instrumento
V1: Fondo Documental	1.1 Bautismo	Nombre	¿Cuál es el nombre del bautizado?	Ficha de Análisis Documental
		Padres	¿Cuáles son los nombres de los padres del bautizado?	Ficha de Análisis Documental
		Padrino	¿Cuáles son los nombres de los padrinos del bautizado?	Ficha de Análisis Documental
		Párroco	¿Cuál es el nombre del párroco que celebro el bautizo?	Ficha de Análisis Documental
		Fecha	¿Qué fecha se bautizó?	Ficha de Análisis Documental
		Lugar de nacimiento	¿Dónde nació?	
	1.2 Matrimonio	Contrayentes	¿Cuál es el nombre de los contrayentes?	Ficha de Análisis Documental
		Padrinos	¿Cuáles son los nombres de los padrinos de los cónyuges?	Ficha de Análisis Documental

Variables	Variable de Interés	Indicadores	Ítems	Instrumento
		Párroco	¿Cuál es el nombre del párroco que celebro el matrimonio?	Ficha de Análisis Documental
		Fecha	¿Qué fecha se celebró el matrimonio?	Ficha de Análisis Documental
	1.3 Defunción	Difunto	¿Cuál es el nombre del difunto?	Ficha de Análisis Documental
		Padres	¿Cuáles son los nombres de los padres del difunto?	Ficha de Análisis Documental
		Párroco	¿Cuál es el nombre del párroco que suscribió la defunción?	Ficha de Análisis Documental
		Fecha	¿Qué fecha se suscribió la defunción?	Ficha de Análisis Documental
	1.4 Confirmación	Nombre	¿Cuál es el nombre del que realizo el sacramento de la confirmación?	Ficha de Análisis Documental
		párroco	¿Qué fecha se celebró el	Ficha de Análisis

Variables	Variable de Interés	Indicadores	Ítems	Instrumento
			sacramento de la confirmación?	Documental
		padrinos	¿Cuáles son los nombres de los padrinos del que recibió el sacramento?	Ficha de Análisis Documental
		padres	¿Cuáles son los nombres de los padres del confirmado?	Ficha de Análisis Documental
		fecha	¿Cuándo se celebró el sacramento de la confirmación?	Ficha de Análisis Documental