

# UNIVERSIDAD SAN PEDRO

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA Y  
DE SISTEMAS



# USP

UNIVERSIDAD SAN PEDRO

## **Sistema informático de gestión sacramental para la iglesia católica San Martín de Porres.**

Tesis para obtener el título profesional de ingeniero en informática  
y de sistemas

**Autor**

Verdeguer Mendoza, Renzo Guillermo

**Asesor**

Ascón Valdivia, Oscar

**Huaraz – Perú**

**2019**

**Palabras clave:**

Tema	Sistema Informático
Especialidad	Ingeniería de software

**Key Words:**

Topic	Information System
Specialization	Software Engineer

**Línea de investigación:**

Área	Ingeniería y tecnología
Sub área	Ingeniería Eléctrica, Electrónica e Informática
Disciplina	Ingeniería de Sistemas y Comunicaciones

**Título**

“SISTEMA INFORMATICO DE GESTION SACRAMENTAL  
PARA LA IGLESIA CATOLICA SAN  
MARTIN DE PORRES”

## **Resumen**

La tesis, tuvo como objetivo desarrollar un sistema informático para la gestión de las actividades eclesíásticas de la Iglesia Católica San Martín de Porres en la ciudad de Huaraz, 2017.

La investigación es de tipo descriptivo–aplicativo no experimental de corte transversal; para el desarrollo del software se utilizó la metodología ágil Scrum, donde se adapta una estrategia de desarrollo incremental, en lugar de una planificación y ejecución completa del aplicativo.

Como resultado del proyecto se tiene un Sistema informático que permita la administración de las actividades eclesíásticas.

## **Abstract**

The thesis, aimed at developing a computer system for the management of the ecclesiastical activities of the Catholic Church San Martin de Porres in the city of Huaraz, 2017.

The research is descriptive-application non-experimental cross-sectional type; For the development of the software we used the agile Scrum methodology, where an incremental development strategy is adapted, instead of a complete planning and execution of the application.

As a result of the project there is a computer system that allows the administration of ecclesiastical activities

## Indice

Palabras clave:.....	I
Título .....	II
Resumen.....	III
Abstract .....	IV
Indice .....	V
1. Introducción .....	1
2. Metodología .....	17
3. Resultados .....	31
Análisis y discusión.....	50
Conclusiones .....	51
Recomendaciones .....	52
Bibliografía .....	53
Agradecimiento.....	55

## 1. Introducción

Las empresas e instituciones en la actualidad se apoyan cada vez más en la tecnología para la mejora de sus procesos y productos. Por lo que la adopción de un sistema web esta dejando de ser una alternativa para pasar a ser un requerimiento en la empresas e instituciones, ya que toda herramienta que ayude a ahorrar tiempo, dinero, recursos y mejorar el servicio a clientes permite lograr la generación de valor para la empresa.

De los antecedentes encontrados se ha abordado los trabajos más relevantes a esta investigación, se tiene una aplicación desarrollado para las iglesias alrededor del mundo denominado **Iglesia Hoy (2001)**, Es un software de escritorio desarrollado en Estados Unidos para iglesias cristianas hispanas alrededor del mundo. Su función es llevar la administración de todas las áreas de la iglesia, como membresías y finanzas, en una forma eficiente. Las ventajas que tiene es el manejo de su propia base de datos local y soportar múltiples usuarios. Y una limitación, es que solo está disponible para sistema operativo Windows.

En la investigación de **Religious Suit (2011)**, Software de escritorio desarrollado en Estados Unidos para la gestión de iglesias lanzado el 2011 para plataformas Windows y Mac. Puede manejar un número ilimitado de miembros. Las ventajas que tiene es ofrecer una gama completa de funcionalidades para la gestión y administración capaz de llevar un registro de miembros, familia, asistencias, comités, finanzas, donaciones, plegarias entre otros y soporta múltiples usuarios.

Así mismo en la aplicación **Membership Librarian (1991)**, Software desarrollado en Estados Unidos en el año 1991, que lleva la cuenta de los miembros de una iglesia, club o asociación, haciendo el seguimiento de los nombres, direcciones, números de teléfonos, direcciones de correo, impuestos, entre otros. Las ventajas que tiene es que incluye un módulo dBase/ASCII de importación/exportación y de copia de seguridad/restauración para guardar sus datos.

Así mismo se revisó la investigación de **Olivo (2014)**, denominada “Sistema web para administración de la Iglesia Cristiana Bautista Sinai de Ibarra, Ecuador” de la

Universidad Técnica del Norte, dentro de la iglesia se manejan varios recursos como recurso humano, de bienes y dinero para lo cual IglesiaSystem es un sistema diseñado para llevar el control de estos recursos, por medio de la asignación de roles a usuarios, mediante un entorno seguro, amigable y confiable. Se Utilizó la metodología RUP para desarrollo de software ya que permite obtener un software de calidad. La implantación del sistema de gestión de la Iglesia SINAI, en internet permitió la automatización de los procesos manuales facilitando el acceso a la información mediante la utilización de tecnología de fácil manejo para los usuarios. El almacenamiento, administración y gestión de la información se realiza a través de base de datos digital, almacenada en un servidor conservando de esta forma la integridad y confidencialidad de la misma, a su vez se restringe el acceso de manipulación de la información sensible de la organización a personal autorizado por el consejo.

En la investigación de **Tovar (2014)**, denominada “Sistema de Información para la gestión de la membresía de la iglesia cristianas”, plantea una estrategia para apoyar la organización y toma de decisiones en las iglesias cristianas con gran número de miembros, que permita implementar los métodos eclesiásticos utilizados mayormente y disminuir la desorganización en la información que presentan muchas instituciones de esta índole. El desarrollo, diseño y la implementación del sistema de información se basan en la metodología RUP (proceso unificado de desarrollo de software). Esta propone tener un proceso iterativo e incremental donde el trabajo se divide en partes más pequeñas permitiendo el equilibrio entre Casos de Uso y Arquitectura, es decir, se centra en la arquitectura de software y está dirigido por casos de uso. Al finalizar el proceso de implementación en esta iglesia, se reflejó un cambio en la organización de la información, la cual paso de ser de muchos papeles a unos formatos digitales accesibles desde cualquier navegador web. Considerando los resultados de la investigación realizada, es factible decir que la utilización del sistema de información para la gestión de membresías en iglesias cristianas sirvió como un gran aporte al manejo y organización de la información en dichas instituciones. Ya que se utilizó la tecnología como apoyo al proceso de toma de decisiones en organizaciones de esta índole.

Del mismo modo se revisó la investigación de **Díaz (2007)**, denominada “Sistema de análisis, diseño e implementación para la administración de comunidades cristianas” de la Universidad Católica del Perú, la finalidad de realizar el sistema para la administración de los principales procesos que tiene una comunidad cristiana de cualquier denominación para lograr mejorar el servicio que tiene la iglesia con sus miembros. Se ha utilizado la metodología RUP, con lenguaje UML 2.0, dado que se advirtió una serie de ventajas, tales como: está orientado a objetos y es iterativo; por lo cual el usuario final, puede tener un avance del sistema y tener observaciones del mismo. Se llegó a contar como producto final, un sistema mediante el cual se podrá trabajar con cualquier estructura celular, adoptada por una iglesia cristiana; gestionando eficazmente los grupos de la iglesia, a fin de mantener la administración de la membresía en forma detallada, obteniendo de manera efectiva, el control en el proceso de células, gestionando las consolidaciones y teniendo una mejor organización de las actividades y ministerios de la iglesia.

Finalmente encontramos un trabajo de tesis de **Pila (2013)**, “Desarrollo de un sistema de administración de las actividades eclesiales de la iglesia católica parroquial Santiago apóstol usando herramientas web”, donde se plantearon el desarrollo de una aplicación Web SISAPE que permite la administración y automatización de procesos y emisión de certificados de bautizo y matrimonio. Se ha seleccionado como metodología de desarrollo a Scrum debido a que incentiva a la participación activa del cliente y que es la que mejor se adapta a las necesidades del proyecto. Como resultado de la investigación el sistema informático permite mejorar la calidad del servicio y facilita el acceso a la información de manera ordenada y clara.

De acuerdo a lo revisado en los antecedentes la investigación se justifica: Socialmente, es relevante porque a través del sistema informático de gestión sacramental se permite ordenar y agilizar las citas eventuales de las actividades sacramentales para que los sacerdotes, monaguillos tengan un mejor control, así mismo la comunidad cristiana se beneficia en un mejor servicio de la iglesia. Respecto al aporte del conocimiento, se automatizó los procesos empleando la metodología ágil scrum, así como herramientas de software con el fin de obtener un producto que

cumple con los procesos de desarrollo de un sistema informático web de control de servicios técnicos para la empresa.

El interés por el estudio se enfocó en la problemática de la iglesia que dentro de las actividades diarias de trabajo que realiza la iglesia, están los registros de eventos sacramentales, cursos de catequesis y confirmación; así como los procesos de emisión de certificados de bautizo y matrimonio, gestión de cursos de catequesis, confirmación y eventos se realizan de forma manual y sin una organización específica lo cual provoca un desorden y desperdicio tanto de tiempo como de recursos económicos, así también los sacerdotes y la secretaria tienen un cruce sobre los horarios hacia las actividades sacramentales, originando la no existencia de una buena gestión hacia los eventos sacramentales, la secretaria no maneja muy bien los sacramentos separados por las personas, así como también los pagos ya que son realizados manualmente. Para tal fin se formuló el problema:

### **¿Cómo desarrollar un sistema informático de gestión sacramental para la Iglesia Católica San Martín de Porres en la ciudad de Huaraz, 2017?**

La investigación es de carácter descriptivo, por lo que se identificó las variables de trabajo: Sistema informático y gestión sacramental.

Respecto a los sistemas informáticos, **Raya J, Raya L y Zurdo J. (2014)**, define como todo sistema, es el conjunto de partes interrelacionadas, hardware, software y de recurso humano (humanware) que permite almacenar y procesar información. El hardware incluye computadoras o cualquier tipo de dispositivo electrónico inteligente, que consisten en procesadores, memoria, sistemas de almacenamiento externo, etc. El software incluye al sistema operativo, firmware y aplicaciones, siendo especialmente importante los sistemas de gestión de bases de datos. Por último, el soporte humano incluye al personal técnico que crean y mantienen el sistema (analistas, programadores, operarios, etc.) y a los usuarios que lo utilizan.

**Desarrollo:** Los sistemas informáticos pasan por diferentes fases en su ciclo de vida, desde la captura de requisitos hasta el mantenimiento. En la actualidad se

emplean numerosos sistemas informáticos en la administración pública, por ejemplo, las operadoras de la policía, el servicio al cliente, entre otras.

**Estructura:** Los sistemas informáticos suelen estructurarse en subsistemas:

- Subsistema físico: asociado al hardware. Incluye entre otros elementos: CPU, memoria principal, placa base, periféricos de entrada y salida, etc.
- Subsistema lógico: asociado al software y la arquitectura; incluye, sistema operativo, firmware, aplicaciones y bases de datos.

**Rouse M. (2016)** define a la gestión del servicio que es un término general que describe un enfoque estratégico para el diseño, la entrega, la gestión y la mejora de la forma en que la tecnología de la información (TI) se utiliza dentro de una organización. El objetivo de cada marco de Gestión de Servicios de TI es asegurar que los procesos adecuados, las personas y la tecnología están en su lugar para que la organización pueda cumplir con sus objetivos de negocio.

### **Metodología Ágil Scrum**

Para automatizar los procesos se empleó la Metodología Ágil Scrum, **Pila (2013)** describe Scrum como un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software dentro del cual las personas pueden afrontar complejos problemas adaptativos, a la vez que entregan productos del máximo valor posible de forma productiva y creativa. Es ligero y fácil de entender.

Scrum es una metodología de desarrollo muy simple, que requiere esfuerzo y compromiso del equipo que forma parte del proyecto, ya que no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias que se presenten durante la evolución del proyecto. Al ser considerada una metodología ágil:

- Es un modelo de desarrollo que se caracteriza por ser más adaptable que predictivo. Está orientado a las personas más que a los procesos.
- Emplea la estructura de desarrollo incremental basada en iteraciones y revisiones periódicas.

- Permite tener un control continuo sobre el estado actual del producto de software.
- Cada iteración termina con un producto de software entregable que se puede publicar o bien seguir mejorando.

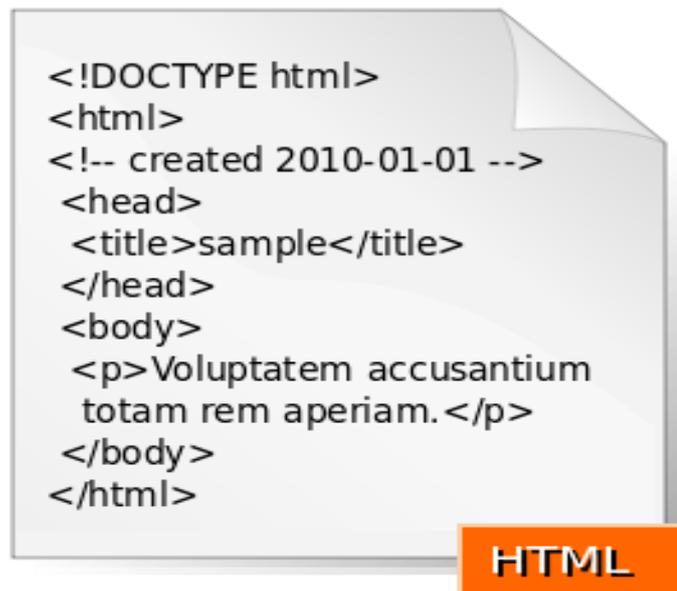
El grado de funcionamiento de Scrum depende directamente de tres condiciones:

1. Características del entorno (organización y proyecto), adecuadas para desarrollo ágil.
2. Conocimiento de la metodología de trabajo por parte del equipo de desarrollo y las personas implicadas por el lado del cliente.
3. Asignación de responsabilidades:
  - Sobre el producto.
  - Sobre el desarrollo.
  - Sobre el funcionamiento de Scrum.

## **Herramientas para el desarrollo del Prototipo**

### **HTML**

HTML es el acrónimo inglés de HyperText Markup Language, que se traduce al español como Lenguaje de Etiquetas de Hipertexto, es un lenguaje de marcado diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web. Este lenguaje de programación se utiliza para el desarrollo de páginas de Internet, se encarga de desarrollar una descripción sobre los contenidos que aparecen como textos, está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla. HTML dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas, tablas y otros, Olivo (2014).



*Figura 1.* Ventana de Trabajo de HTML  
Fuente: Libro Curso Completo de HTML

## CSS

Que en español significa hojas de estilo en cascada, es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o como se va a imprimir, o incluso como va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura, se utiliza para dar estilo a documento HTML y XML. Funciona en base a reglas o declaraciones que tiene dos partes: un selector y la declaración, y a su vez la declaración está compuesta por una propiedad y el valor que se le asigne. El W3C es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación, Olivo (2014).

```
2 body {
3   background-attachment: scroll;
4   background-color: #FF8080;
5   background-position: inherit;
6   border-color: #FFFFFF;
7   border-style: dotted;
8   color: #000000;
9   float: left;
10  font-family: Arial, Helvetica, sans-serif;
11  font-size: medium;
12  font-style: normal;
13  font-variant: normal;
14  font-weight: normal;
15  height: auto;
16  letter-spacing: normal;
17  line-height: normal;
18 }
```

*Figura 2.* Ventana de Trabajo de CSS  
Fuente: CSS Avanzado por Javier Eguiluz

## JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario, técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java, Olivo (2014).

```
server.js x
4
5 var database = require('./database');
6
7 var server = express();
8 server.use(bodyParser.json());
9
10 server.use(function (req, res, next) {
11
12 // allow origin for demo purposes
13 res.setHeader('Access-Control-Allow-Origin', 'http://localhost:8080');
14 res.setHeader('Access-Control-Allow-Methods', 'GET, POST, DELETE');
15 res.setHeader('Access-Control-Allow-Headers', 'X-Requested-With, content-type');
16 next();
17 });
18
19 // Routes
```

server.js server - 7 references

```
var bodyParser = require('body-parser');
var _ = require('lodash');
var database = require('./database');
var server = express();
server.use(bodyParser.json());
server.use(function (req, res, next) {
  server.get('/todos', fu...
  server.post('/todos', f...
  server.delete('/todos/...
  server.listen(PORT, fu...
```

Figura 3. Ventana de trabajo de JavaScript  
Fuente: Elaboración Propia

## JQUERY

Es una rápida, pequeña y altamente rica librería de JavaScript open source, que funciona en múltiples navegadores, y que es compatible con CSS. Su objetivo principal es hacer la programación scripting mucho más fácil y rápida del lado del cliente. Con JQuery se pueden producir paginas dinámicas, así como animaciones parecidas a flash en relativamente corto tiempo.

La ventaja principal de JQuery es que es mucho más fácil que sus competidores. Usted puede agregar plugins fácilmente, traducándose esto en un ahorro substancial de tiempo y esfuerzo. De hecho, una de las principales razones por la cual Resig y su equipo crearon JQuery fue para ganar tiempo (en el mundo de desarrollo web, tiempo importa mucho).

En resumen:

- jQuery es flexible y rápido para el desarrollo web
- Viene con licencia MIT y es Open Source
- Tiene una excelente comunidad de soporte
- Tiene Plugins
- Bugs son resueltos rápidamente

- Excelente integración con AJAX

Poniendo ventajas y desventajas a un lado, el futuro de jQuery se ve sumamente prometedor. Definitivamente es una de las mejores librerías en el mundo JavaScript. Y claramente, al final del día, jQuery es como cualquier otra herramienta, que será tan buena como tan bueno sea el programador. Olivo (2014).

```

jQuery.fn = jQuery.prototype = {
  constructor: jQuery,
  init: function( selector, context, rootjQuery ) {
    var match, elem, ret, doc;

    // Handle $(""), $(null), $(undefined), $(false)
    if ( !selector ) {
      return this;
    }

    // Handle $(DOMElement)
    if ( selector.nodeType ) {
      this.context = this[0] = selector;
      this.length = 1;
      return this;
    }

    // Handle HTML strings
    if ( typeof selector === "string" ) {
      if ( selector.charAt(0) === "<" && selector.charAt( selector.length - 1 ) === ">" && selector.length >= 3 ) {
        // Assume that strings that start and end with <> are HTML and skip the regex check
        match = [ null, selector, null ];
      } else {
        match = rquickExpr.exec( selector );
      }

      // Match html or make sure no context is specified for #id

```

Figura 4. Ventana de Trabajo de JQUERY  
Fuente: Elaboración propia

## MySQL Base de Datos

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo<sup>1 2</sup>, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

MySQL fue inicialmente desarrollado por MySQL AB (empresa fundada por David Axmark, Allan Larsson y Michael Widenius). MySQL A.B. fue adquirida por Sun Microsystems en 2008, y ésta a su vez fue comprada por Oracle Corporation en 2010, la cual ya era dueña desde 2005 de Innobase Oy, empresa finlandesa desarrolladora del motor InnoDB para MySQL, Olivo (2014).

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de doble licenciamiento anteriormente mencionado. La base de datos se distribuye en varias versiones, una Community, distribuida bajo la Licencia pública general de GNU, versión 2, y varias versiones Enterprise, para aquellas empresas que quieran incorporarlo en productos privativos. Las versiones Enterprise incluyen productos o servicios adicionales tales como herramientas de monitorización y soporte oficial. En 2009 se creó un fork denominado MariaDB por algunos desarrolladores (incluido algunos desarrolladores originales de MySQL) descontentos con el modelo de desarrollo y el hecho de que una misma empresa controle a la vez los productos MySQL y Oracle Database, Olivo (2014).

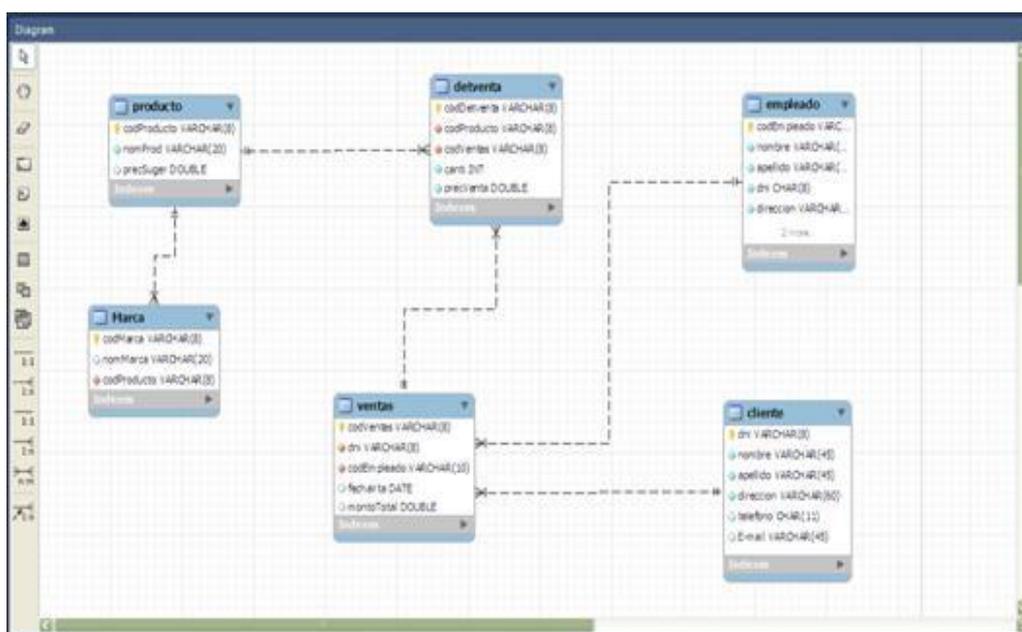


Figura 5. Ventana de Trabajo de MySQL  
Fuente: Elaboración Propia

## **WampServer**

Según (Ivan, 2013), Wampserver es un entorno de desarrollo web para Windows con el que podrás crear aplicaciones web con Apache, PHP y bases de datos MySQL database. También incluye PHPMyAdmin y SQLiteManager para manejar tus bases de datos.

**Características:** Provee a los desarrolladores con los cuatro elementos necesarios para un servidor web: un Sistema Operativo (Windows), un manejador de base de datos (MySQL), un software para servidor web (Apache) y un software de programación script Web (PHP (generalmente), Python o PERL), debiendo su nombre a dichas herramientas. Lo mejor de todo es que WAMP5 es completamente gratuito. WAMP incluye, además de las últimas versiones de Apache, PHP y MySQL, versiones anteriores de las mismas, para el caso de que se quiera testear en un entorno de desarrollo particular.

**Utilidad:** El uso de WAMP permite servir páginas HTML a Internet, además de poder gestionar datos en ellas, al mismo tiempo WAMP, proporciona lenguajes de programación para desarrollar aplicaciones Web.

### **Requerimiento de wampserver**

#### 1. Del lado del servidor:

- Windows, como sistema operativo;
- Apache, como servidor web;
- MySQL, como gestor de bases de datos;
- PHP (generalmente), Perl, o Python, como lenguajes de programación.

#### 2. Del lado del Cliente:

- Navegador de Internet

## Funcionalidades de WampServer

WampServer tiene funcionalidades que lo hacen muy completo y fácil de usar. Con un click izquierdo sobre el icono de WampServer, usted será capaz de:

- Gestionar sus servicios de Apache y MySQL,
- Cambiar de línea / fuera de línea (dar acceso a todos o sólo local)
- Instalar y cambiar de Apache, MySQL y PHP emisiones
- Gestión de la configuración de sus servidores
- Acceder a sus registros
- Acceder a sus archivos de configuración
- Crear alias

## Lenguajes de programación

- HTML
- JavaScript
- CSS
- PHP
- ASP
- ASP.NET
- JSP
- Python
- RUBY
- APACHE

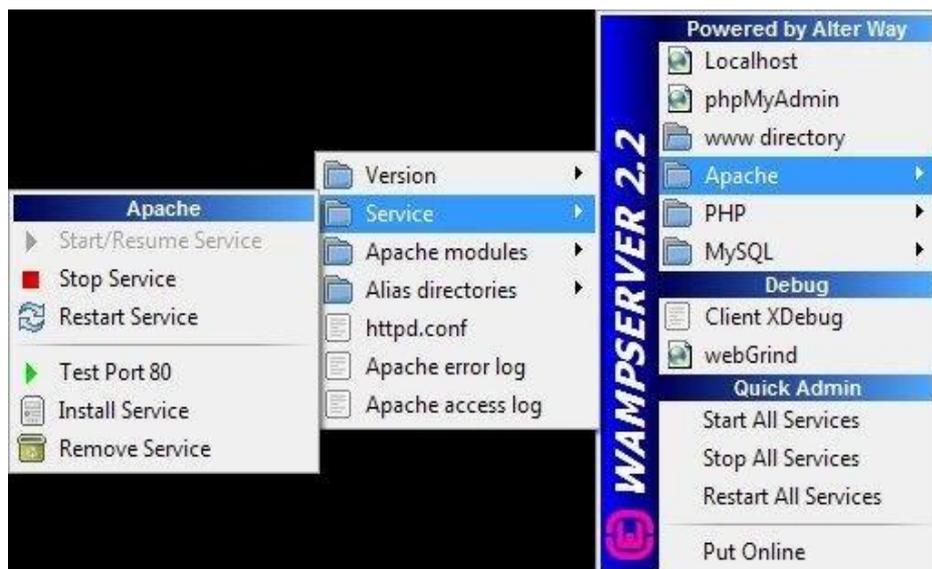


Figura 6. Ventana de trabajo de WampServer  
Fuente: Elaboración propia

## PHP

La plataforma de desarrollo web PHP (Hypertext Pre-processor) es el nombre de un lenguaje de programación muy potente que, junto con html, permite crear sitios web dinámicos. Php se instala en el servidor y funciona con versiones de Apache, Microsoft IIs, Netscape Enterprise Server y otros, para IglesiaSystem utilizaremos Apache La forma de usar php es insertando código php dentro del código html de un sitio web. Cuando un cliente (cualquier persona en la web) visita la página web que contiene éste código, el servidor lo ejecuta y el cliente sólo recibe el resultado. Su ejecución, es por tanto en el servidor, a diferencia de otros lenguajes de programación que se ejecutan en el navegador. Puede ser ejecutado en la mayoría de los sistemas operativos (Windows, Mac OS, Linux, Unix), Olivo (2014).

```
37 private function getData() {
38     $data = new stdClass();
39
40     $results = $this->mysql->query( 'SELECT * FROM hit_counter' );
41     if ( $results->rowCount() == 0 ) {
42
43
44         $data->total = 0;
45         $data->unique = 0;
46
47         $stmt = $this->mysql->prepare( 'INSERT INTO hit_counter( 'total_hits', 'unique_hits' ) VALUES(:total, :unique)' );
48         $stmt->bindParam( ':total', $data->total );
49         $stmt->bindParam( ':unique', $data->unique );
50         $stmt->execute();
51
52     } else {
53         $rows = $results->fetchAll( PDO::FETCH_OBJ );
54         $data->total = $rows[0]->total_hits;
55         $data->unique = $rows[0]->unique_hits;
56     }
57
58     return $data;
59 }
60
61 private function isNewVisitor() {
62     return array_key_exists( 'visited', $_SESSION ) && $_SESSION['visited'] != true;
63 }
64
65 private function visit() {
66
67     $this->mysql->query( "UPDATE hit_counter SET total_hits = total_hits + 1" );
68
69     if ( $this->isNewVisitor() ) {
70         $this->mysql->query( "UPDATE hit_counter SET unique_hits = unique_hits + 1";
71         $_SESSION['visited'] = true;
72     }
73 }
```

Figura 7. Ventana de Trabajo de PHP

Fuente: Elaboración propia

## **Sublime Text**

Sublime Text es un editor de código multiplataforma, ligero y con pocas concesiones a las florituras. Es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente.

Sublime Text permite tener varios documentos abiertos mediante pestañas, e incluso emplear varios paneles para aquellos que utilicen más de un monitor. Dispone de modo de pantalla completa, para aprovechar al máximo el espacio visual disponible de la pantalla.

El programa cuenta “de serie” con 22 combinaciones de color posibles, aunque se pueden conseguir más. Para navegar por el código cuenta con Minimap, un panel que permite moverse por el código de forma rápida.

El sistema de resaltado de sintaxis de Sublime Text soporta un gran número de lenguajes (C, C++, C#, CSS, D, Erlang, HTML, Groovy, Haskell, HTML, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, Matlab, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML).

El programa dispone de auto-guardado, muchas opciones de personalización, cuenta con un buen número de herramientas para la edición del código y automatización de tareas. Soporta macros, Snippets y auto completar, entre otras funcionalidades. Algunas de sus características son ampliables mediante plugins, Benítez (2012).

```
4 $sticky = get_option( 'sticky_pos' );
5
6
7 if ( empty( $sticky ) ) {
8     return $sticky;
9 } else {
10     $sticky_args = array(
11         'posts_per_page' => 1,
12         'post_in' => $sticky,
13         'ignore_sticky_posts' => 1
14     );
15     $sticky_query = new WP_Query( $
16     return $sticky_query;
17 }
18 }
19 }
20 }
21 }
22 }
23 }
```

Figura 8. Ventana de Trabajo de Sublime Text  
Fuente: Elaboración Propia

La investigación tiene un alcance de carácter descriptivo, no es posible plantear una hipótesis debido a que no se intenta correlacionar o explicar casualidad de variables y el objetivo a alcanzar está claro. Por tal razón se considera una hipótesis implícita.

El objetivo General del presente estudio es: Desarrollar un sistema informático de gestión sacramental para la Iglesia Católica San Martín de Porres – Huaraz, 2017 y como objetivos específicos:

- Establecer los procesos para la administración de las actividades eclesiales para la Iglesia Católica San Martín de Porres
- Aplicar la metodología ágil scrum para el sistema informático
- Construir el sistema utilizando el lenguaje HTML, CSS y JavaScript

## 2. Metodología

La presente investigación es de carácter descriptivo; porque la recopilación de datos obtenidos por instrumentos de investigación nos permitió observar, conocer y describir la situación en la que se encuentra la empresa.

El diseño de la investigación es no experimental de corte transversal por que los datos serán tomados en una sola vez utilizando los instrumentos de recolección de datos.

Se determinó emplear Scrum como metodología Ágil para el desarrollo de un Sistema Informático de gestión sacramental para la Iglesia Católica San Martin de Porres Huaraz,2017.

Observando a empresas como Honda, HP, Canon... etc., se dieron cuenta de que el producto no seguía unas fases en las que había un equipo especializado en cada una de ellas, si no que se partía de unos requisitos muy generales y el producto lo realizaba un equipo multidisciplinar que trabajaba desde el comienzo del proyecto hasta el final.

Jeff Sutherland y Ken Schwaber (1996) presentaron las prácticas que se usaban como proceso formal para el desarrollo de software y que pasarían a incluirse en la lista de Agile Alliance.

Luis Chacón y Huber Sinche (2013), nos indica que Scrum al ser una metodología de desarrollo ágil tiene como base la idea de creación de ciclos breves para el desarrollo, que comúnmente se llaman iteraciones y que en Scrum se llamarán “Sprints”.

Scrum aparece como una práctica destinada a los productos tecnológicos y será en 1993 cuando realmente Jeff Sutherland aplique un modelo de desarrollo de Software en Ease/Corporation.

Según (Manuel, 2012), Scrum es adecuado para aquellas empresas en las que el desarrollo de los productos se realiza en entornos que se caracterizan por tener:

**Incertidumbre**: Sobre esta variable se plantea el objetivo que se quiere alcanzar sin proporcionar un plan detallado del producto.

Esto genera un reto y da una autonomía que sirve para generar una “tensión” adecuada para la motivación de los equipos.

**Auto-organización:** Los equipos son capaces de organizarse por sí solos, no necesitan roles para la gestión, pero tienen que reunir las siguientes características:

- **Autonomía:** Son los encargados de encontrar la solución usando la estrategia que encuentren adecuada.
- **Autosuperación:** Las soluciones iniciales sufrirán mejoras.
- **Auto-enriquecimiento:** Al ser equipos multidisciplinares se ven enriquecidos de forma mutua, aportando soluciones que puedan complementarse.

**Control moderado:** Se establecerá un control suficiente para evitar descontrol. Se basa en crear un escenario de “autocontrol entre iguales” para no impedir la creatividad y espontaneidad de los miembros del equipo.

**Transmisión del conocimiento:** Todo el mundo aprende de todo el mundo. Las personas pasan de unos proyectos a otros y así comparten sus conocimientos a lo largo de la organización.

Scrum al ser una metodología de desarrollo ágil tiene como base la idea de creación de ciclos breves para el desarrollo, que comúnmente se llaman *iteraciones* y que en Scrum se llamarán “*Sprints*”.

Para entender el ciclo de desarrollo de Scrum es necesario conocer las 5 fases que definen el ciclo de desarrollo ágil:

**Concepto:** Se define de forma general las características del producto y se asigna el equipo que se encargará de su desarrollo.

**Especulación:** en esta fase se hacen disposiciones con la información obtenida y se establecen los límites que marcarán el desarrollo del producto, tales como costes y agendas.

Se construirá el producto a partir de las ideas principales y se comprueban las partes realizadas y su impacto en el entorno.

Esta fase se repite en cada iteración y consiste, en rasgos generales, en:

- Desarrollar y revisar los requisitos generales.
- Mantener la lista de las funcionalidades que se esperan.
- Plan de entrega. Se establecen las fechas de las versiones, hitos e iteraciones. Medirá el esfuerzo realizado en el proyecto.

**Exploración:** Se incrementa el producto en el que se añaden las funcionalidades de la fase de especulación.

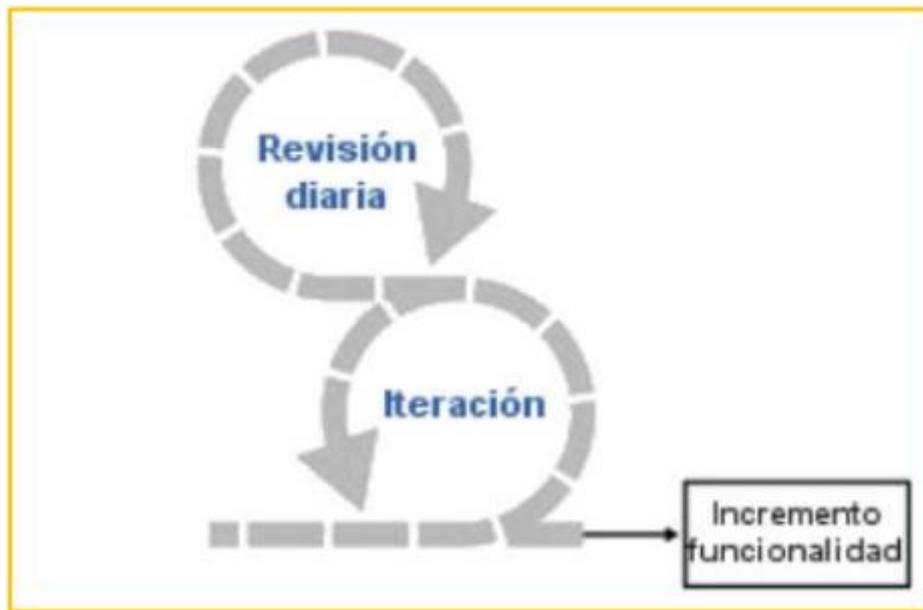
**Revisión:** El equipo revisa todo lo que se ha construido y se contrasta con el objetivo deseado.

**Cierre:** Se entregará en la fecha acordada una versión del producto deseado. Al tratarse de una versión, el cierre no indica que se ha finalizado el proyecto, sino que seguirá habiendo cambios, denominados “mantenimiento”, que hará que el producto final se acerque al producto final deseado.



*Figura 9.* Ciclo de Desarrollo ágil  
Fuente: Metodología Scrum

Scrum gestiona estas iteraciones a través de reuniones diarias, uno de los elementos fundamentales de esta metodología.



*Figura 10.* Ciclo Principal de Scrum  
Fuente: Metodología Scrum

## Componentes

Para entender todo el proceso de desarrollo del Scrum, se describirá de forma general las fases y los roles. Estas fases y roles se detallarán de forma más concisa más adelante.

Scrum se puede dividir de forma general en 3 fases, que podemos entender como reuniones. Las reuniones forman parte de los artefactos de esta metodología junto con los roles y los elementos que lo forman.

### a) Reuniones

- **Planificación del Backlog**

Se definirá un documento en el que se reflejarán los requisitos del sistema por prioridades.

En esta fase se definirá también la planificación del Sprint 0, en la que se decidirá cuáles van a ser los objetivos y el trabajo que hay que realizar para esa iteración.

Se obtendrá además en esta reunión un Sprint Backlog, que es la lista de tareas y que es el objetivo más importante del Sprint.

- **Seguimiento del Sprint**

En esta fase se hacen reuniones diarias en las que las 3 preguntas principales para evaluar el avance de las tareas serán:

- ¿Qué trabajo se realizó desde la reunión anterior?
- ¿Qué trabajo se hará hasta una nueva reunión?
- Inconvenientes que han surgido y qué hay que solucionar para poder continuar.

- **Revisión del Sprint**

Cuando se finaliza el Sprint se realizará una revisión del incremento que se ha generado. Se presentarán los resultados finales y una demo o versión, esto ayudará a mejorar el feedback con el cliente.

## b) Roles

Los roles se dividen en 2 grupos: cerdos y gallinas, esto surge en el chiste sobre un cerdo y una gallina y su intención de poner un restaurante.



Figura 11. El Cerdo y la Gallina  
Fuente: Metodología Scrum

- **Los Cerdos**

Son las personas que están comprometidas con el proyecto y el proceso de Scrum.

- **Product Owner:** Es la persona que toma las decisiones, y es la que realmente conoce el negocio del cliente y su visión del producto. Se encarga de escribir las ideas del cliente, las ordena por prioridad y las coloca en el Product Backlog.
- **ScrumMaster:** Es el encargado de comprobar que el modelo y la metodología funciona. Eliminará todos los inconvenientes que hagan que el proceso no fluya e interactuará con el cliente y con los gestores.
- **Equipo De Desarrollo:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint. Los equipos son auto-manejados y auto-organizados, y son responsables de convertir, al finalizar la iteración, el conjunto de requerimientos del Sprint Backlog en un incremento del producto final. Los miembros del equipo son colectivamente responsables del éxito o fracaso de cada iteración y del proyecto como un todo.

- **Las Gallinas**

Aunque no son parte del proceso de Scrum, es necesario que parte de la retroalimentación dé la salida del proceso y así poder revisar y planear cada sprint.

- **Usuarios:** Es el destinatario final del producto.
- **Stakeholders:** Las personas a las que el proyecto les producirá un beneficio. Participan durante las revisiones del Sprint.
- **Managers:** Toma las decisiones finales participando en la selección de los objetivos y de los requisitos.

### c) Elementos

Los elementos que forman a Scrum son:

- **Product Backlog:** lista de necesidades del cliente.
- **Sprint Backlog:** lista de tareas que se realizan en un Sprint.
- **Incremento:** parte añadida o desarrollada en un Sprint, es una parte terminada y totalmente operativa.

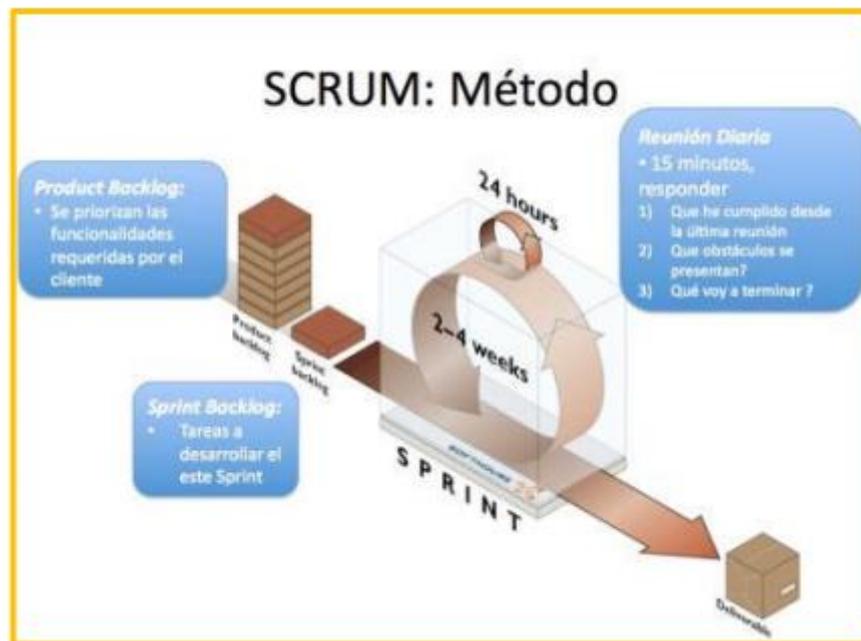


Figura 12. Ciclo de Desarrollo Scrum  
Fuente: Metodología Scrum

- **Product Backlog**

Es el inventario en el que se almacenan todas las funcionalidades o requisitos en forma de lista priorizada. Estos requisitos serán los que tendrá el producto o los que irá adquiriendo en sucesivas iteraciones.

La lista será gestionada y creada por el cliente con la ayuda del Scrum Master, quien indicará el coste estimado para completar un requisito, y además contendrá todo lo que aporte un valor final al producto.

Las tres características principales de esta lista de objetivos serán:

- Contendrá los objetivos del producto, se suele usar para expresarlos las historias de usuario.
- En cada objetivo, se indicará el valor que le da el cliente y el coste estimado; de esta manera, se realiza la lista, priorizando por valor y coste, se basará en el ROI.
- En la lista se tendrán que indicar las posibles iteraciones y los realces que se han indicado al cliente.
- La lista ha de incluir los posibles riesgos e incluir las tareas necesarias para solventarlos.

Es necesario que antes de empezar el primer Sprint se definan cuáles van a ser los objetivos del producto y tener la lista de los requisitos ya definida. No es necesario que sea muy detallada, simplemente deberá contener los requisitos principales para que el equipo pueda trabajar. Realizar este orden de tareas tiene como beneficios:

- El proyecto no se paraliza simplemente por no tener claro los requisitos menos relevantes, y el cliente podrá ver resultados de forma más rápida.
- Los requisitos secundarios aparecerán a medida que se va desarrollando el proyecto, por lo tanto, no se pierde tanto tiempo en analizarlos al principio y el cliente será más consciente de sus necesidades.
- Los requisitos secundarios puede que no se lleguen a necesitar porque se han sustituido o porque no reportan un retorno ROI interesante.

Una vez definidos los requisitos se tendrá que acordar cuándo se tiene que entender un objetivo como terminado o completado.

Se entiende que un producto está completado si:

- Asegura que se puede realizar un entregable para realizar una demostración de los requisitos y ver qué se han cumplido.
- Incluirá todo lo necesario para indicar que se está realizando el producto que el cliente desea.

Como complemento a la definición de completado, se debería de asociar una condición de aceptación o no aceptación a cada objetivo en el mismo momento en el que se crea la lista.

Finalmente, el Product Backlog irá evolucionando mientras el producto exista en el mercado. Esta es la forma para evolucionar y tener un valor de producto para el cliente suficiente para ser competitivo.

- **Las historias de Usuario.**

Son las descripciones de las funcionalidades que va a tener el software.

Estas historias de usuario serán el resultado de la colaboración entre el cliente y el equipo, e irán evolucionando durante toda la vida del proyecto.

Las historias de usuario se componen de tres fases denominadas “Las 3 C”:

- Card: Será una breve descripción escrita que servirá como recordatorio.
- Conversation: Es una conversación que servirá para asegurarse de que se ha entendido bien todo, y concretar el objetivo.
- Confirmation: Tests funcionales para fijar detalles que sean relevantes e indicar cuál va a ser el límite.

En cuanto al formato, un modelo podría ser como el que se muestra en la siguiente imagen:

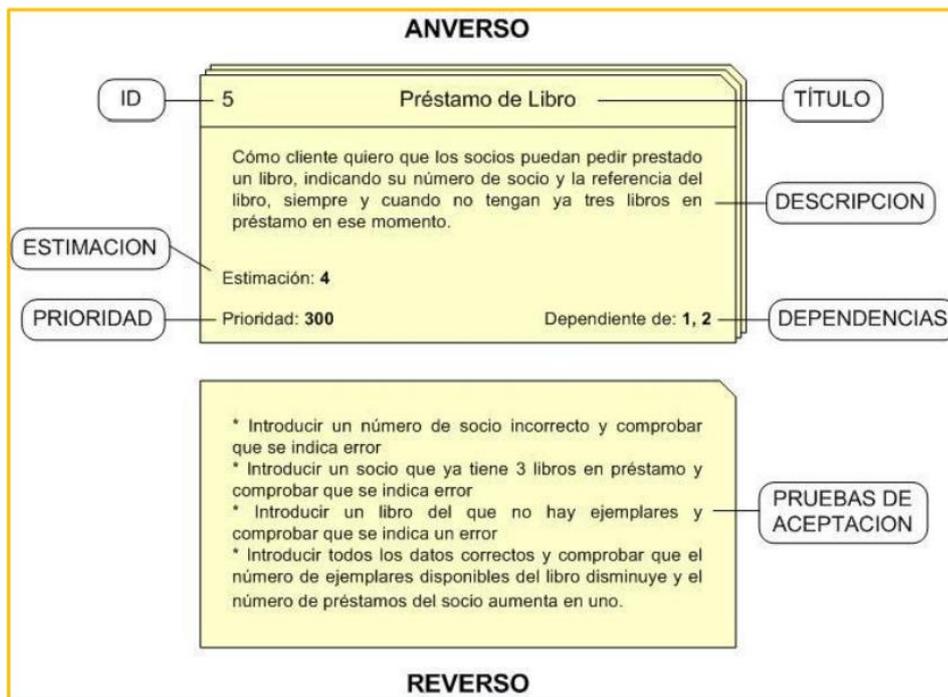


Figura 13. Ciclo de Desarrollo Scrum  
Fuente: Metodología Scrum

**ID:** Identificador de la historia de usuario.

**TÍTULO:** Título descriptivo de la historia de usuario.

**DESCRIPCIÓN:** Descripción sintetizada de la historia de usuario.

**ESTIMACIÓN:** Evaluación del coste de implementación en unidades de desarrollo. Estas unidades representarán el tiempo teórico (desarrollo/hombre) que se haya estimado al comienzo del proyecto.

**PRIORIDAD:** Prioridad en la implementación de la historia de usuario respecto al resto de las historias de usuario. A mayor número, mayor prioridad. Otra aproximación a la priorización de tareas se hace a través del método MoSCoW:

**M – Must,** se debe completar este requerimiento para finalizar el proyecto.

**S – Should,** se debe completar este proyecto por todos los medios, pero el éxito del proyecto no depende de él.

**C–Could**, se debería completar este requerimiento si su implementación no afecta a la consecución de los objetivos principales del proyecto.

W – Would, se puede completar este requerimiento si sobra tiempo de desarrollo (o en futuras versiones del mismo)

**DEPENDENCIAS:** Una historia de usuario no debería ser dependiente de otra historia, pero a veces es inevitable. En este apartado se indicarían los IDs de las tareas de las que depende una tarea.

▪ **Formato de la Pila Del Producto (Product Backlog).**

En Scrum, la preferencia por tener documentación en todo momento es menos estricta. Se encuentra más necesario el mantener una comunicación directa con el equipo, por eso se usa como herramienta el Backlog.

Aunque no hay ningún producto especial a la hora de confeccionar la lista, es conveniente que incluya información relativa a:

- Identificador para la funcionalidad.
- Descripción de la funcionalidad.
- Sistema de priorización u orden.
- Estimación.

Id	Prioridad	Descripción	Est.	Por
1	Muy alta	Plataforma tecnológica	30	AR
2	Muy alta	Interfaz usuario	40	LR
3	Muy alta	Un usuario se registra en el sistema	40	LR
4	Alta	El operador define el flujo y textos de un expediente	60	AR
5	Alta	Etc...	999	XX

Figura 14. Ciclo de desarrollo Scrum  
Fuente: Metodología Scrum

- **Sprint backlog**

Es la lista de tareas que elabora el equipo durante la planificación de un Sprint. Se asignan las tareas a cada persona y el tiempo que queda para terminarlas.

De esta manera el proyecto se descompone en unidades más pequeñas y se puede determinar o ver en qué tareas no se está avanzando e intentar eliminar el problema.

Requisito	Tarea	Quien	Estado (No iniciada / en progreso / completada)	Dia:												
				1	2	3	4	5	6	7	8	9	10			
				Horas	1120	1088	1076	1048	1040	1032	1020	1008	992	972		
				pendientes												
Requisito A	Tarea 1	Joao	Completada	16	8											
Requisito A	Tarea 4	Laura	Completada	4												
Requisito A	Tarea 5	Laura	Completada	4												
Requisito A	Tarea 3	Gabri	Completada	8												
Requisito A	Tarea 2	Laura	Completada	16	8	4										
Requisito A	Tarea 6	Gabri	Completada	8	8	8										
Requisito A	Tarea 7	Joao	Completada	16	16	16	8									
Requisito A	Tarea 8	Laura	Completada	8	8	8										
Requisito A	Tarea 9	Laura	Completada	8	8	8	8	8								
Requisito A	Tarea 10	Laura	Completada	8	8	8	8	8	8	4						
Requisito A	Tarea 11	Joao	Completada	16	16	16	16	16	16	8						
Requisito B	Tarea 12	Gabri	Completada	16	16	16	16	16	16	16	16	8				
Requisito B	Tarea 13	Laura	Completada	16	16	16	16	16	16	16	16	8				
Requisito B	Tarea 14	Joao	En progreso	8	8	8	8	8	8	8	8	8	4			
Requisito B	Tarea 15	Gabri	En progreso	8	8	8	8	8	8	8	8	8	8			
Requisito B	Tarea 16	Laura	En progreso	8	8	8	8	8	8	8	8	8	8			
Requisito C	Tarea 17	Joao	No iniciada	4	4	4	4	4	4	4	4	4	4			
Requisito C	Tarea 18	Gabri	No iniciada	8	8	8	8	8	8	8	8	8	8			
Requisito C	Tarea 19	Laura	No iniciada	16	16	16	16	16	16	16	16	16	16	16		
Requisito C	Tarea 20	Joao	No iniciada	8	8	8	8	8	8	8	8	8	8			

Figura 15. Sprint Backlog  
Fuente: Metodología Scrum

Cómo funciona la lista:

- Es una lista ordenada por prioridades para el cliente.
- Puede haber dependencias entre una tarea y otra, por lo tanto, se tendrá que diferenciar de alguna manera.
- Todas las tareas tienen que tener un coste semejante que será entre 4-16 horas.

Formato de lista:

- Hojas de cálculo.
- Pizarras.
- Herramientas colaborativas.

Generalmente, las tareas a completar se suelen gestionar mediante el Scrum Taskboard, a cada objetivo se le asignan las tareas necesarias para llevarlo a cabo, se usan post-its que se van moviendo de una columna a otra para cambiar el estado.

Se debe incluir:

- Lista de tareas.
- Persona responsable de cada tarea, el estado en el que se encuentra y el tiempo que queda por terminarla.
- Permite la consulta diaria del equipo.
- Permite tener una referencia diaria del tiempo que le queda a cada tarea.

- **Incremento**

Representa los requisitos que se han completado en una iteración y que son perfectamente operativos.

Según los resultados que se obtengan, el cliente puede ir haciendo los cambios necesarios y replanteando el proyecto.

#### **d) Beneficios**

- Unión de equipo: Fomenta el trabajo en equipo, focalizando todos los esfuerzos en alcanzar un objetivo común.
- Cumplimiento de expectativas: Cliente establece sus expectativas indicando el valor que le aporta cada requisito, equipo los estima y el Product Owner establece su prioridad.
- Flexibilidad a cambios: Capacidad de reacción ante los cambios de requerimientos por necesidades del cliente o evoluciones del mercado.
- Reducción del Time to Market: El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de finalizar por completo.
- Mayor calidad de software: Necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- Mayor productividad: Motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- Predicciones de tiempo: Se conoce la velocidad media del equipo por sprint, con lo que es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad.
- Reducción de riesgos: Al realizar las funcionalidades de más valor en primer lugar y conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

### 3. Resultados

Para el informe de investigación se han aplicado dos encuestas, las cuales se han orientado a los expertos en temas de desarrollo software nacional y a los usuarios finales los cuales serán beneficiados con el desarrollo. Aplicada las encuestas a todos los expertos y usuarios finales del proceso control de servicios técnicos, donde se obtuvieron los siguientes resultados:

#### Resultados de Encuesta Aplicada a Expertos

1. ¿Qué gestor de base de datos es recomendable para el sistema?

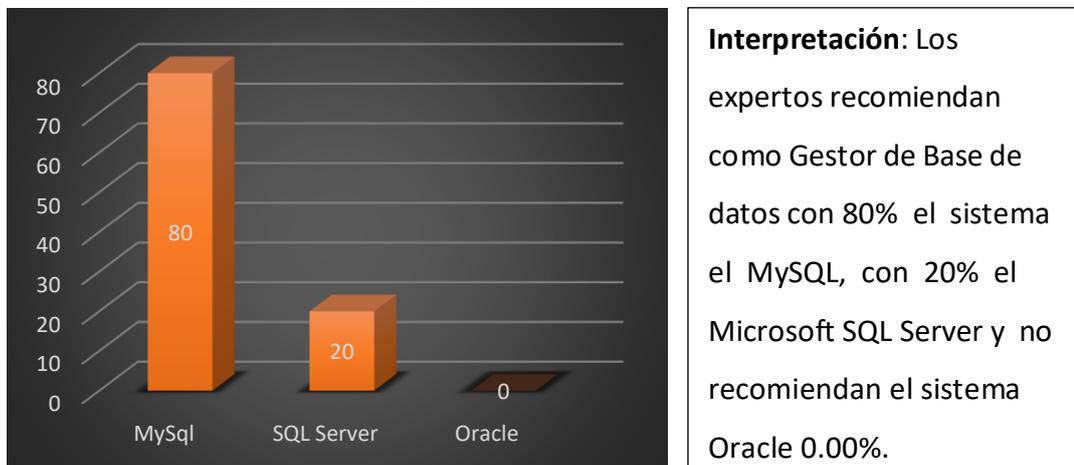


Figura 16. Recomendación de Gestor de Base de Datos  
Fuente: Elaboración Propia

2. ¿Qué lenguaje de programación es recomendable? (Sistema informático).

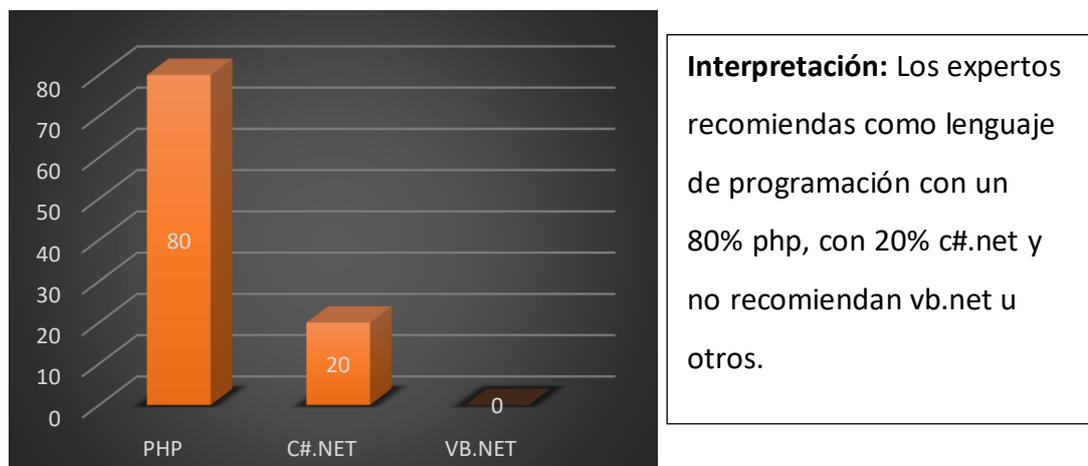


Figura 17. Recomendación de Lenguaje de Programación  
Fuente: Elaboración Propia

### 3. Metodología de Desarrollo: Siendo un proyecto de poca duración (4 meses)

¿Que metodología de Desarrollo sería recomendable?

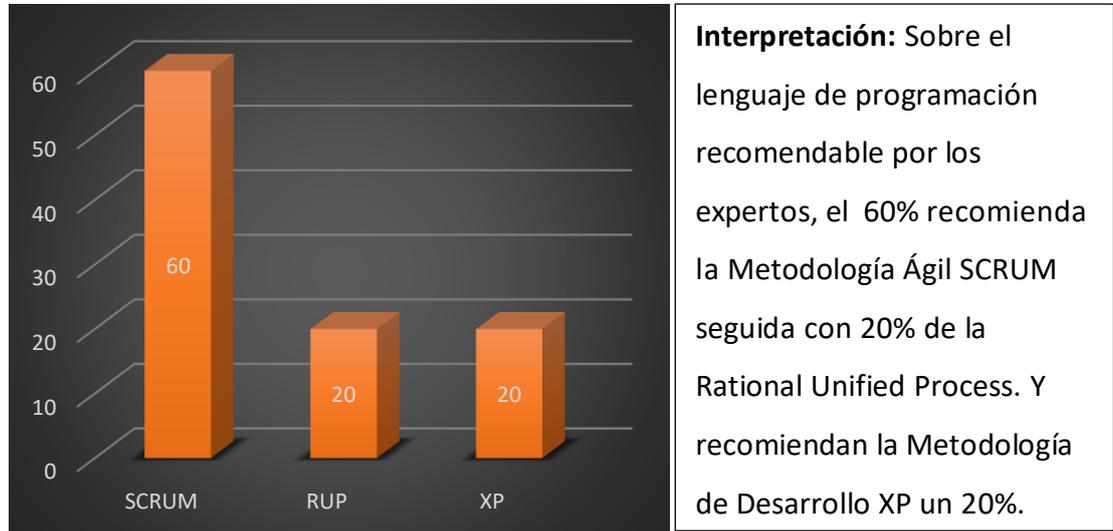


Figura 18. Recomendación Metodología de Desarrollo a Utilizar  
Fuente: Elaboración Propia

## Resultados de Encuesta Aplicada a Usuarios

### 4. ¿Cómo considera la calidad de procesos de la Iglesia actualmente?

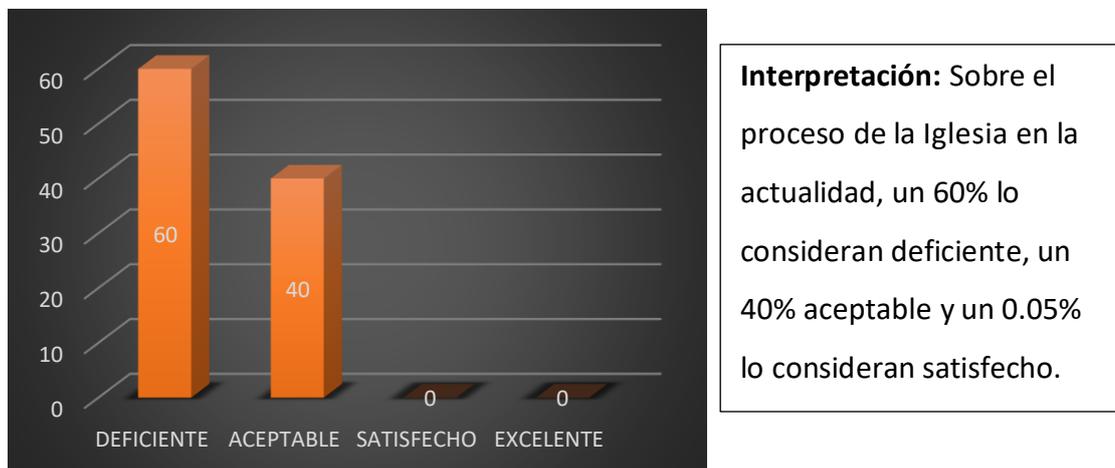
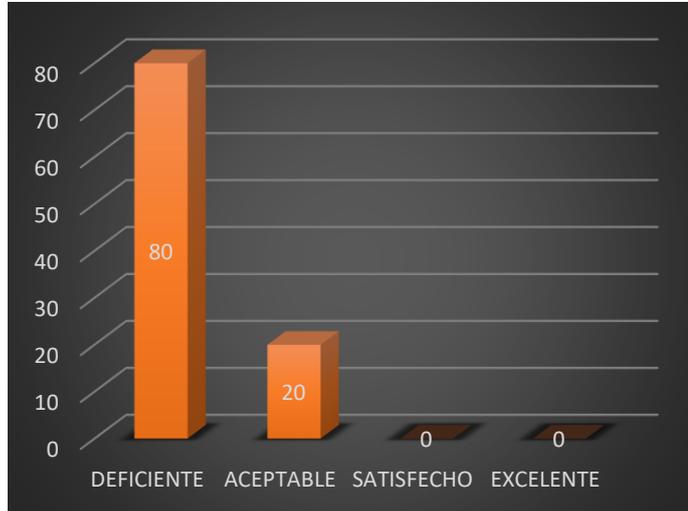


Figura 19. Calidad de los Procesos en la Empresa  
Fuente: Elaboración propia

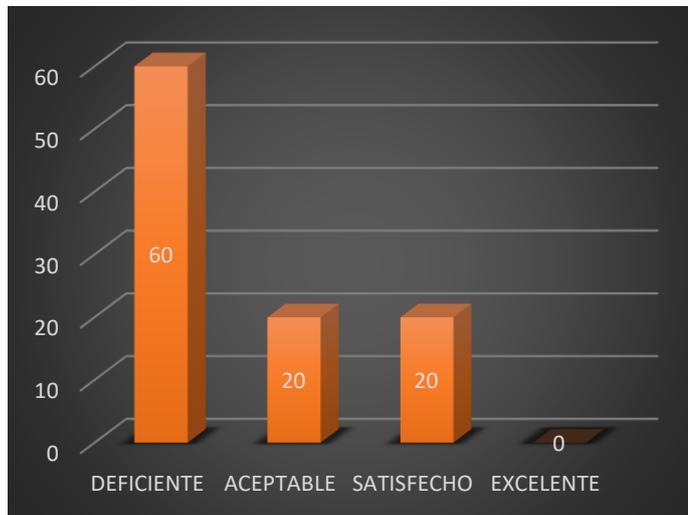
5. ¿Cree que la información obtenida por la Iglesia es buena para la toma de decisiones?



**Interpretación:** El 80% de trabajadores indican que la información obtenida para la toma de decisiones es deficiente y el 20 %

Figura 20. Toma de Decisiones Respecto a la Información  
Fuente: Elaboración propia

6. ¿Cómo considera los reportes que maneja actualmente?



**Interpretación:** Sobre los reportes que maneja la empresa actualmente, el 60% lo consideran deficiente, el 20% lo considera aceptable y 20%

Figura 21. Calificación respecto al servicio  
Fuente: Elaboración propia

7. ¿Cómo califica usted que los servicios tengan control por medio de un sistema informático?

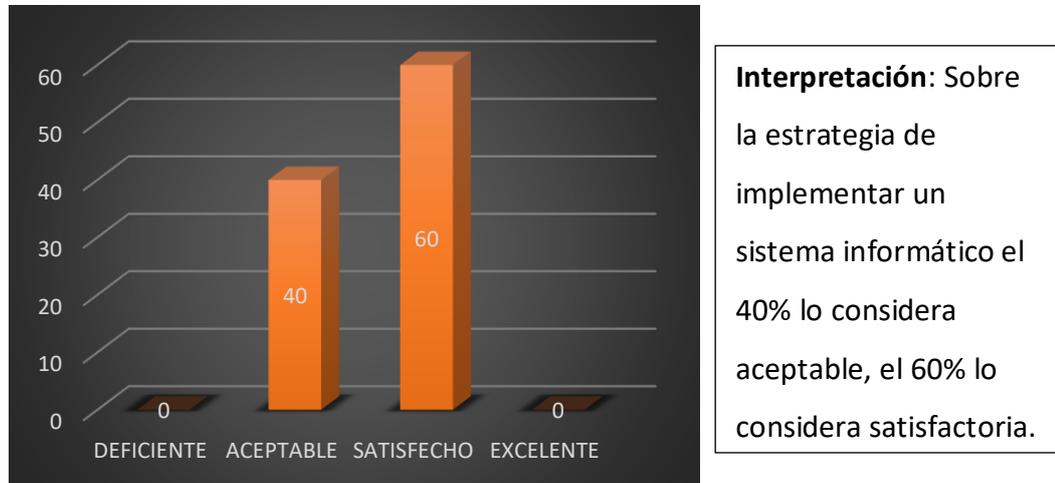


Figura 22. Calificación respecto al servicio  
Fuente: Elaboración

### Aplicación de la Metodología Ágil SCRUM

Las principales razones del uso de un ciclo de desarrollo iterativo e incremental de tipo scrum para la ejecución de este proyecto son:

- Sistema modular. Las características del sistema [**Nombre del sistema**] permiten desarrollar una base funcional mínima y sobre ella ir incrementando las funcionalidades o modificando el comportamiento o apariencia de las ya implementadas.
- Entregas frecuentes y continuas al cliente de los módulos terminados, de forma que puede disponer de una funcionalidad básica en un tiempo mínimo y a partir de ahí un incremento y mejora continua del sistema.
- Previsible inestabilidad de requisitos.
  - Es posible que el sistema incorpore más funcionalidades de las inicialmente identificadas.
  - Es posible que durante la ejecución del proyecto se altere el orden en el que se desean recibir los módulos o historias de usuario terminadas.

- Para el cliente resulta difícil precisar cuál será la dimensión completa del sistema, y su crecimiento puede continuarse en el tiempo suspenderse o detenerse.

### **Presentación del equipo de trabajo y asignación de roles**

En la siguiente tabla se presenta el equipo y los roles de trabajo.

*Tabla 1. Asignación de Roles*

ROLES	PERSONA ASIGNADA	DESCRIPCION
<b>Product Owner</b>	Padre Juri Liñan	Representante de la parte del negocio, se encarga de dar las historias de usuario.
<b>Proxy Product Owner</b>	Juan Torres	Hereda los deberes del producto owner, se encarga de actualizar el producto backlog y verificar los entregables.
<b>Scrum Master</b>	Renzo Verdeguer	Encargado de poner en práctica la metodología y ayuda a gestionar la reducción de impedimentos.

La metodología Scrum permite al cliente participar activamente en el proyecto y lo anima a brindar sus ideas permitiendo obtener un producto de software que tenga el mayor nivel de satisfacción posible.

### **Planificación de un Sprint**

La reunión de planificación de cada Sprint se hizo el primer día de cada uno, a las 9:15 de la mañana, y en un principio se pensaba que duraría toda la jornada de trabajo, aunque finalmente la duración fue variable dependiendo del Sprint.

### **Sprints Diarios (Daily Scrum)**

Se definió un lugar y una hora para las reuniones diarias, serían a las 9:15 de la mañana en la iglesia, y su duración de 15 minutos como máximo.

En estas reuniones cada miembro del equipo explicaba en qué estaba trabajando, qué había terminado y si tenía algún problema. A raíz de estos comentarios se

producían debates y se tomaban decisiones sobre cómo abordar los distintos problemas que se estuviesen produciendo.

Al terminar la reunión, el Scrum Master actualizaba completamente la pizarra.

### **Elaboración del Product backlog**

Para desarrollar el sistema de administración de las actividades eclesióásticas de la Iglesia Católica San Martín de Porres es necesario conocer los requerimientos y características del sistema desde el punto de vista de cliente, por lo que se procede a elaborar el Product Backlog, en el cual cada ítem corresponde a un requerimiento del cliente. Se establece la prioridad y la estimación de tiempo en la que se planea entregar cada requerimiento; y en base a esta estimación se define el sprint en el que se realizara cada ítem.

La prioridad la establece el cliente y se basa en una escala del 1 al 5, siendo 5 la menor prioridad y 1 la prioridad más alta.

*Tabla 2.* Product Backlog

<b>ID</b>	<b>CARACTERISTICAS/FUNCIONALIDAD</b>	<b>IMPORTANCIA</b>
<b>1</b>	Administrar Usuarios	Alta
<b>2</b>	Administrar Personal	Alta
<b>3</b>	Administrar tipo de Sacramento	Alta
<b>4</b>	Registro de eventos sacramentales	Alta
<b>5</b>	Gestión de requisitos sacramentales	Alta
<b>6</b>	Listado de eventos	Alta

### **Planificación de cada Sprint**

La reunión de planificación de cada Sprint se la realiza al inicio de los mismos. Se elabora el Sprint Backlog correspondiente, donde se especifica qué ítems del Product Backlog se desarrollarán y las tareas que se realizarán para cumplir con cada uno de éstos. Se utiliza para saber el estado y avance de cada una de las tareas, lo que permite tomar decisiones para conseguir que los objetivos de cada sprint se cumplan en los plazos estimados.

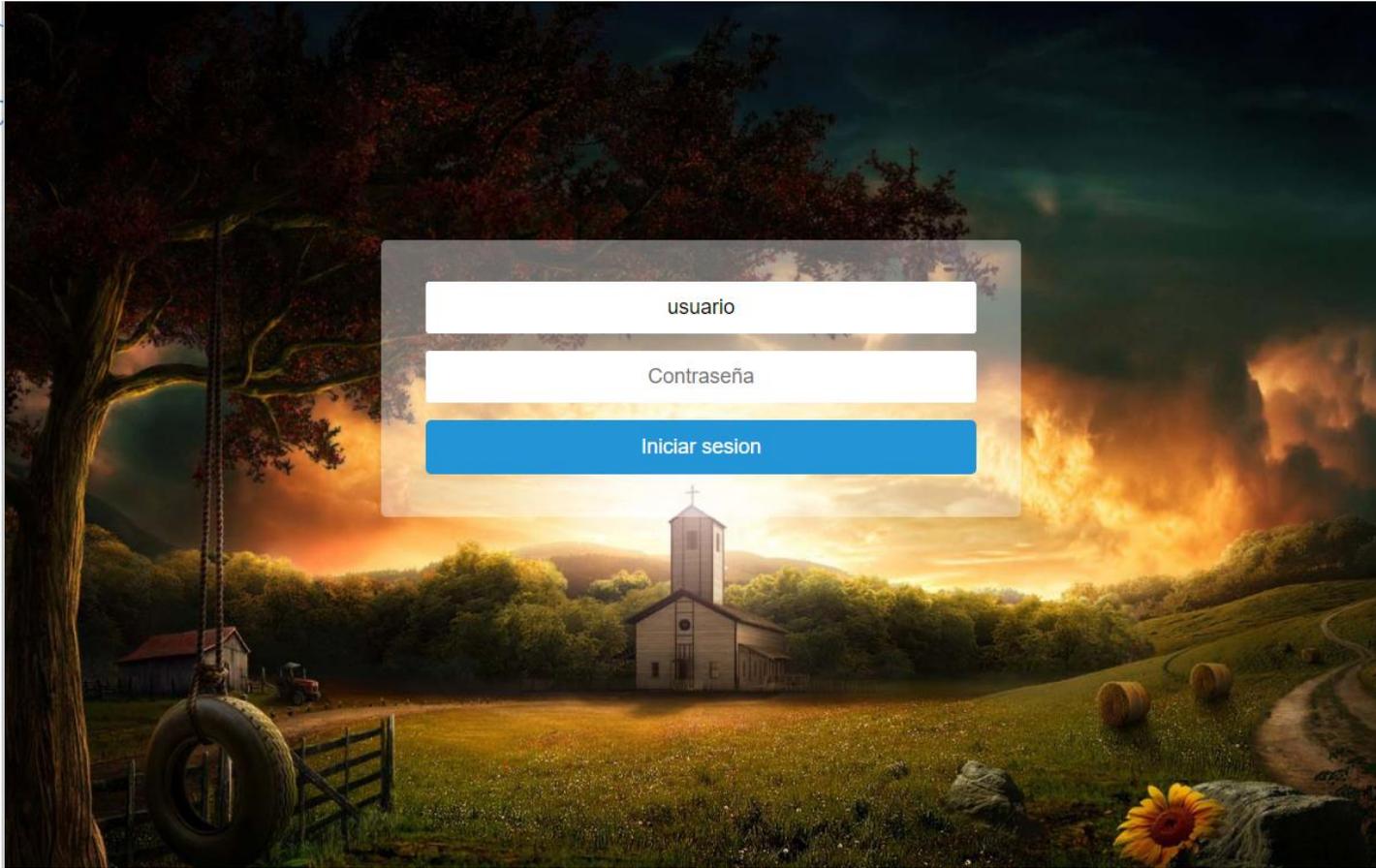
Este documento está compuesto por el Id del ítem del Product Backlog, las tareas de cada ítem, el Id de cada tarea, la duración estimada en horas y el estado.

El Sprint Backlog obtenido para los Sprint se muestra en las siguientes tablas.

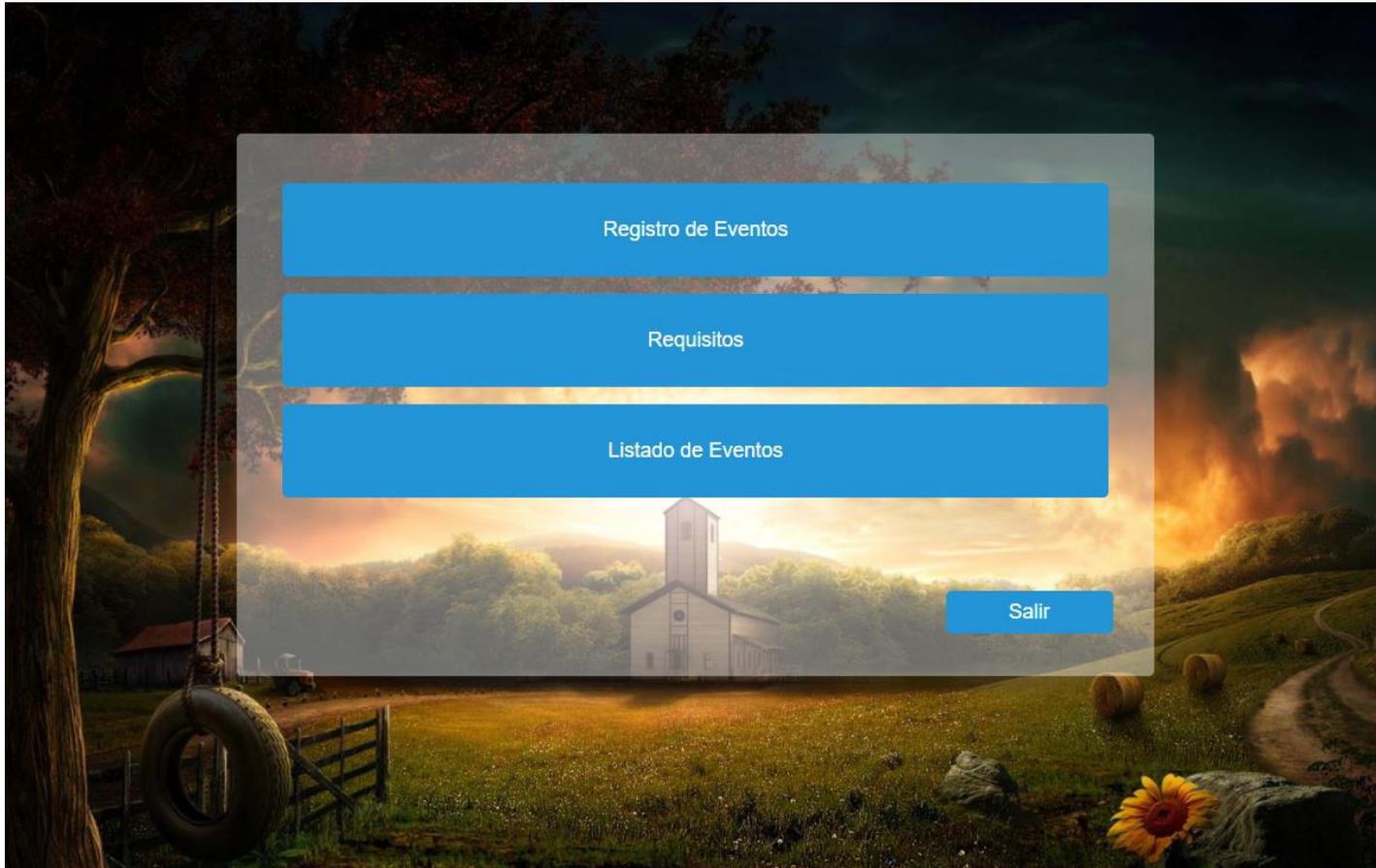
Tabla 3. Sprint Backlog 1

SPRINT	INICIO	DURACIÓN	OBJETIVO
01	Lun 22/05/17	52 h.	Administrar Usuario, se crea la interfaz y la programación de los usuarios para el sistema con los parámetros contraseña, username y se asigna perfiles de usuario según se requiera.
		<b>Horas pendientes</b>	288
		<b>Tareas pendientes</b>	24

BACKLOG ID: 01- Análisis del DCN y al área de Matemáticas.					
Tarea	Categoría	Estimado en Horas	Estado	Responsable	Esfuerzo 0-10
Análisis del modelo conceptual correspondiente a usuarios.	Análisis	12	Terminado	Renzo	4/10
Diseño de la interfaz gráfica de la creación de un nuevo usuario.	Diseño	24	Terminado	Renzo	6/10
Programación de la lógica de negocio correspondiente a Usuarios.	Programación	14	Terminado	Juan	3/10
Pruebas de aceptación.	Análisis/Diseño	2	Terminado	Juan	2/10



*Figura 23.* Interfaz Grafica de Ingreso de Usuario  
Fuente: Elaboración propia



*Figura 24.* Menú  
Fuente: Elaboración propia

Tabla 4. Sprint Backlog 2

SPRINT	INICIO	DURACIÓN	OBJETIVO			
02	Lun 29/05/17	40 h.	Administrar Personal, se crea la interfaz y la programación de los personales (Registro, modificación y eliminación).			
		<b>Horas pendientes</b>	236			
		<b>Tareas pendientes</b>	20			
BACKLOG ID: 01- Análisis del DCN y al área de Matemáticas.						
Tarea	Categoría	Estimado en Horas	Estado	Responsable	Esfuerzo 0-10	
Análisis del modelo conceptual correspondiente a los personales	Análisis	12	Terminado	Renzo	4/10	
Diseño de la interfaz gráfica de la creación de un nuevo personal.	Diseño	24	Terminado	Renzo	6/10	
Programación de la lógica de negocio correspondiente al personal.	Programación	2	Terminado	Juan	3/10	
Pruebas de aceptación.	Análisis/Diseño	2	Terminado	Juan	2/10	

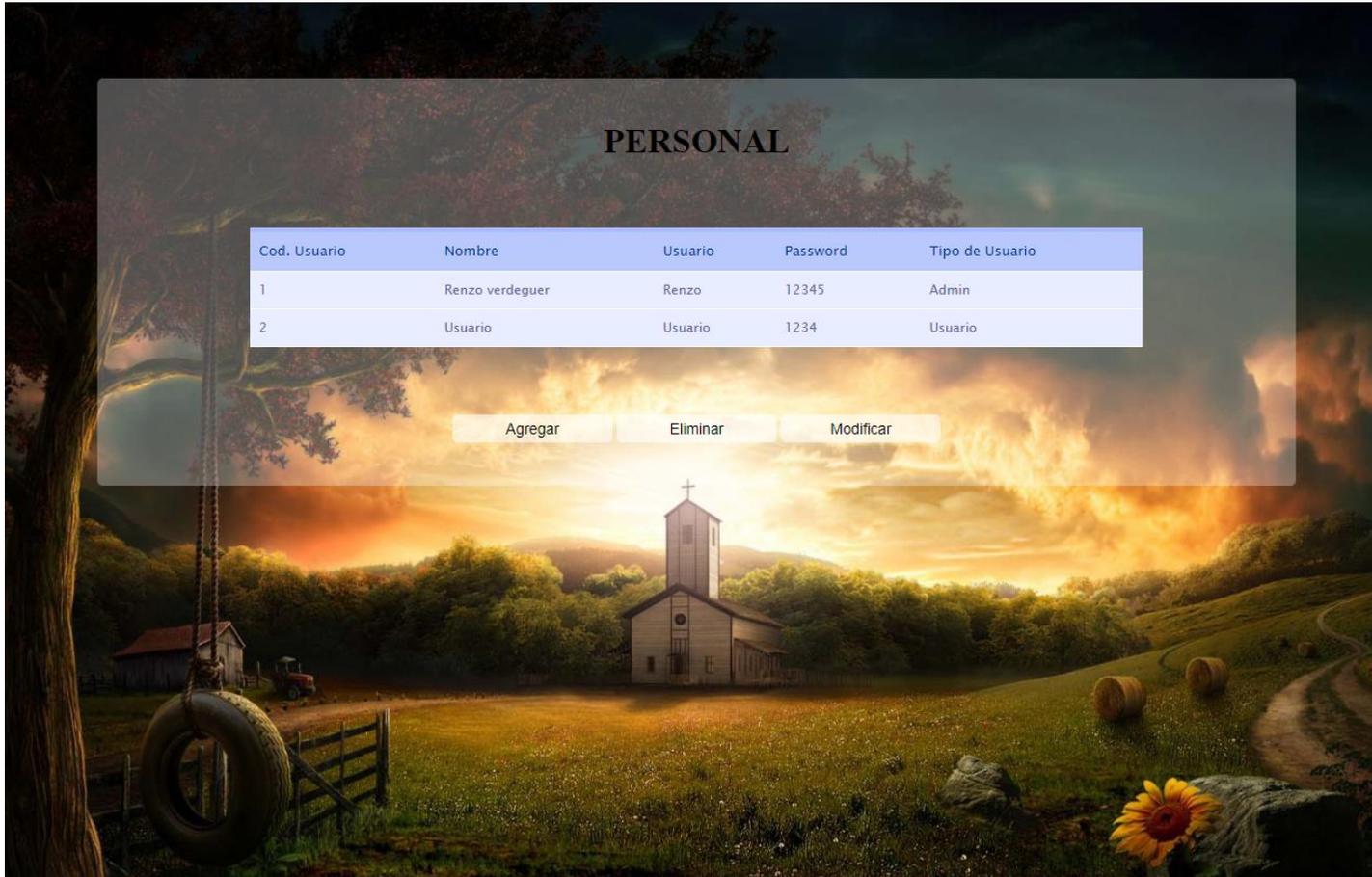


Figura 25. Administración de Personal  
Fuente: Elaboración Propia

Tabla 5. Sprint Backlog 3

<b>SPRINT</b>	<b>INICIO</b>	<b>DURACIÓN</b>	<b>OBJETIVO</b>			
<b>03</b>	<b>Lun 05/06/17</b>	<b>52 h.</b>	Administrar tipos de sacramentos, se crea la interfaz y la programación de los tipos de sacramento (Bautizo, Matrimonio, Comunión, entre otros ).			
		<b>Horas pendientes</b>	196			
		<b>Tareas pendientes</b>	16			
<b>BACKLOG ID: 01- Análisis del DCN y al área de Matemáticas.</b>						
<b>Tarea</b>	<b>Categoría</b>	<b>Estimado en Horas</b>	<b>Estado</b>	<b>Responsable</b>	<b>Esfuerzo 0-10</b>	
Análisis del modelo conceptual correspondiente al tipo de sacramento	Análisis	12	Terminado	Renzo	4/10	
Diseño de la interfaz gráfica para la elección del sacramento.	Diseño	24	Terminado	Renzo	6/10	
Programación de la lógica de negocio correspondiente a la elección del sacramento.	Programación	14	Terminado	Juan	3/10	
Pruebas de aceptación.	Análisis/Diseño	2	Terminado	Juan	2/10	

## REGISTRO DE PARTICIPANTE

Seleccionar Sacramento: Bautizo ▼

### Informacion del Niño

Nombre del Niño :

fecha de Nacimiento:

Sexo: Masculino ▼

### Informacion de los Padres

Nombre del Padre :

Nombre de la Madre :

Direccion:

Telefono:

### Informacion del Padrino

Nombre del Padrino :

Direccion del Padrino :

Telefono:

### Informacion de la Madrina

Nombre del Madrina :

Direccion de la Madrina :

Telefono:

Registrar Regresar

Figura 26. Administración de personal  
Fuente: Elaboración Propia

Tabla 6. Sprint Backlog 4

<b>SPRINT</b>	<b>INICIO</b>	<b>DURACIÓN</b>	<b>OBJETIVO</b>			
<b>04</b>	<b>Mie 15/06/17</b>	<b>58 h.</b>	Registros Sacramentales, se crea la interfaz, programación y las reglas sobre los registros sacramentales			
		<b>Horas pendientes</b>	138			
		<b>Tareas pendientes</b>	12			
<b>BACKLOG ID: 01- Análisis del DCN y al área de Matemáticas.</b>						
<b>Tarea</b>	<b>Categoría</b>	<b>Estimado en Horas</b>	<b>Estado</b>	<b>Responsable</b>	<b>Esfuerzo 0-10</b>	
Análisis del modelo conceptual correspondiente a la solicitud de los cristianos para los sacramentos.	Análisis	12	Terminado	Renzo	4/10	
Diseño de la interfaz gráfica de la creación de un nuevo evento sacramental.	Diseño	24	Terminado	Renzo	6/10	
Programación de la lógica de negocio correspondiente a los eventos sacramentales.	Programación	20	Terminado	Juan	3/10	
Pruebas de aceptación.	Análisis/Diseño	2	Terminado	Juan	2/10	

**REGISTRO DE PARTICIPANTE**

Seleccionar Sacramento:

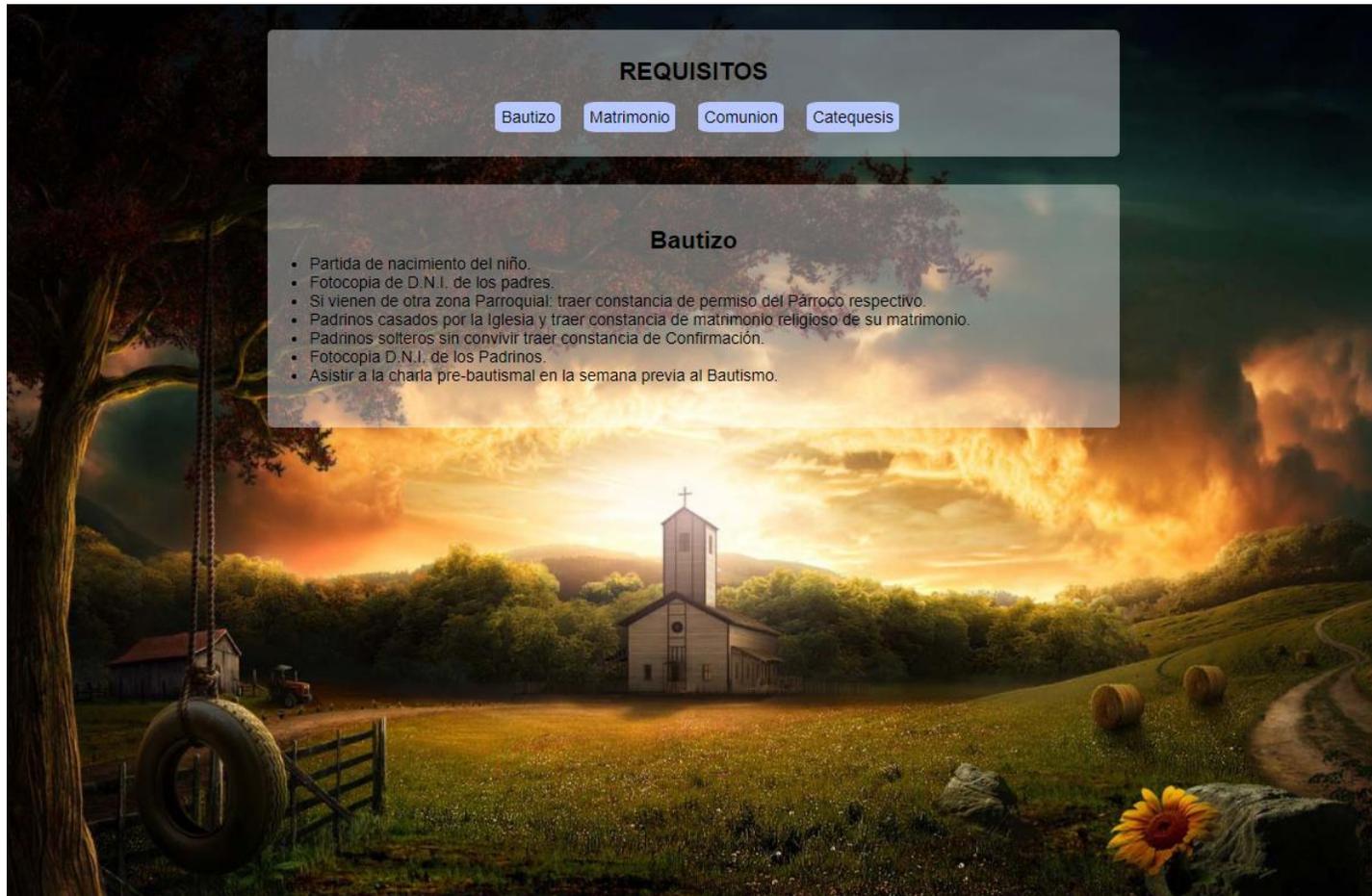
Informacion del Niño	Informacion de los Padres
Nombre del Niño : <input type="text"/>	Nombre del Padre : <input type="text"/>
fecha de Nacimiento: <input type="text" value="dd/mm/aaaa"/>	Nombre de la Madre : <input type="text"/>
Sexo: <input type="text" value="Masculino"/>	Direccion: <input type="text"/>
	Telefono: <input type="text"/>

Informacion del Padrino	Informacion de la Madrina
Nombre del Padrino : <input type="text"/>	Nombre del Madrina : <input type="text"/>
Direccion del Padrino : <input type="text"/>	Direccion de la Madrina : <input type="text"/>
Telefono: <input type="text"/>	Telefono: <input type="text"/>

Figura 27. Registro de un Evento Sacramental  
Fuente: Elaboración propia

Tabla 7. Sprint Backlog 5

<b>SPRINT</b>	<b>INICIO</b>	<b>DURACIÓN</b>	<b>OBJETIVO</b>			
<b>05</b>	<b>Mar 20/06/17</b>	<b>46 h.</b>	Gestión de requisitos sacramentales, se crea la interfaz, programación y la información sobre los requisitos de cada sacramento.			
		<b>Horas pendientes</b>	92			
		<b>Tareas pendientes</b>	8			
<b>BACKLOG ID: 01- Análisis del DCN y al área de Matemáticas.</b>						
<b>Tarea</b>	<b>Categoría</b>	<b>Estimado en Horas</b>	<b>Estado</b>	<b>Responsable</b>	<b>Esfuerzo 0-10</b>	
Análisis del modelo conceptual correspondiente a los requisitos sacramentales.	Análisis	12	Terminado	Renzo	4/10	
Diseño de la interfaz gráfica de la sobre los requisitos sacramentales	Diseño	24	Terminado	Renzo	6/10	
Programación de la lógica de negocio correspondiente a los requisitos sacramentales.	Programación	8	Terminado	Juan	3/10	
Pruebas de aceptación.	Análisis/Diseño	2	Terminado	Juan	2/10	



*Figura 28.* Requisitos de los Sacramentos  
Fuente: Elaboración Propia

Tabla 8. Sprint Backlog 6

SPRINT	INICIO	DURACIÓN	OBJETIVO			
06	jue 22/05/17	52 h.	Listado de eventos, se crea la interfaz y programación para los eventos registrados.			
		<b>Horas pendientes</b>	52			
		<b>Tareas pendientes</b>	4			
BACKLOG ID: 01- Análisis del DCN y al área de Matemáticas.						
Tarea	Categoría	Estimado en Horas	Estado	Responsable	Esfuerzo 0-10	
Análisis del modelo conceptual correspondiente al listado de eventos registrado.	Análisis	12	Terminado	Renzo	4/10	
Diseño de la interfaz gráfica sobre el listado de los eventos	Diseño	24	Terminado	Renzo	6/10	
Programación de la lógica de negocio correspondiente a listados de eventos	Programación	2	Terminado	Juan	3/10	
Pruebas de aceptación.	Análisis/Diseño	2	Terminado	Juan	2/10	

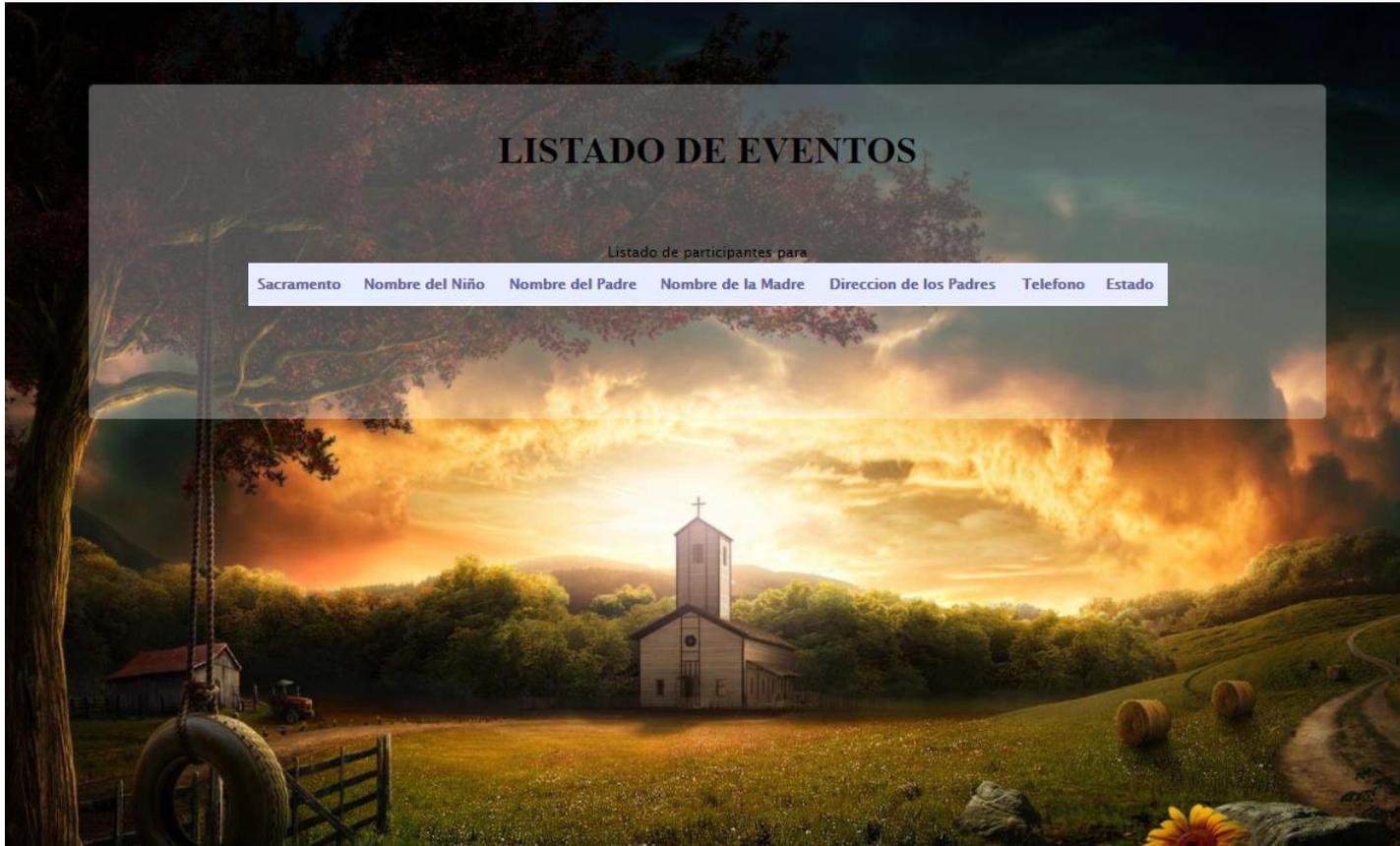


Figura 29. Listado de Eventos  
Fuentes: Elaboración Propia

## **Análisis y discusión**

El documento generado servirá como fuente de consulta para estudiantes y profesionales de ingeniería que requieran formular una propuesta en el campo de los sistemas informáticos con diversas herramientas aplicadas en la gestión de información de organizaciones y de proyectos de la institución. Mediante el desarrollo del proyecto se evidencia la vinculación de los estudiantes de Ingeniería de la Universidad San Pedro, como ente que propone cambios e innovación tecnológica, permitiendo poner en práctica los conocimientos adquiridos en beneficio de la colectividad.

En los antecedentes de mi investigación e considerado a los siguientes trabajos como:

Pila (2013), este antecedente me permitió usar la guía de la metodología que eh empleado en mi investigación para poder compararla con la mía ya que me permitió utilizar las mismas herramientas.

El aporte que obtuvimos fue la obtención exacta de todos los requerimientos necesarios. Asimismo, me ayudo a decidir en utilizar la metodología ágil Scrum describiendo como utilizar de forma efectiva los Sprints y manejar las distintas reuniones.

Iglesia Hoy (2000), Es un software ya desarrollado para administrar las actividades de distintas iglesias alrededor del mundo, tiene una buena interfaz que se acopla a la mayoría de iglesia para su administración.

El análisis de los resultados...

Lo que tomamos en cuenta fue la visualización sobre su interfaz dándome ideas más claras para poder incorporarlas a mi investigación.

Olivo (2014), Las conclusiones por las que opte en investigar este antecedente fue por la herramienta y los distintos lenguajes de programación que utiliza.

En la presente investigación me sirvió a tomar, elegir las herramientas a usar, así como los lenguajes para la programación.

## Conclusiones

- Al ser un Sistema informático está diseñada para permitir a futuro la entrega de nuevos servicios a la comunidad, a los que se podrá acceder usando un navegador.
- La metodología Scrum se caracteriza por ser incremental e iterativa, permitiendo una mayor participación del cliente en el proyecto y una interacción continua con el equipo de desarrollo, para obtener un producto de software de calidad y que cumpla con sus expectativas.
- Los cambios en los requerimientos solicitados por el cliente se realizaron con éxito y sin grandes impactos en el sistema, lo que se logró gracias a que Scrum sugiere la entrega de pequeñas versiones del sistema al final de cada sprint para que sean analizadas y aprobadas por el cliente.
- El uso de software libre en el presente proyecto permite que el equipo de desarrollo se enfoque entregar un producto de software con la mayor calidad posible sin preocuparse por el costo de licencias y permisos.
- La selección de las herramientas de desarrollo apropiadas para el presente proyecto permite que se aprovechen las capacidades, conocimiento y experiencia de la desarrolladora, proporcionándole un mejor entendimiento, habilidad y práctica al usarlas.

## **Recomendaciones**

- Se recomienda el uso de software libre para reducir los costos en un proyecto de desarrollo de software.
- El uso de la metodología Scrum en proyectos de desarrollo de software es altamente recomendable, ya que garantiza que el cliente obtenga un producto de calidad, en el tiempo establecido y que cumpla los objetivos propuestos.
- Se recomienda ampliar la zona de cobertura de la red inalámbrica existente en la Iglesia Católica San Martín de Porres, para que el sistema brinde un mejor servicio y esté disponible a la comunidad.
- Establecer un horario fijo para las reuniones diarias que dure aproximadamente 15 minutos, así el equipo tendrá una mejor comunicación y se organizarán las tareas oportunamente.
- Se recomienda especificar normas y estándares para el ingreso de la información en el sistema y para su gestión en general dentro de la Iglesia, con el objetivo de evitar errores y futuros inconvenientes por un manejo inadecuado.

## Bibliografía

- Aguirre y Sinche (2013), *Diseño de una aplicación móvil para la consulta académica de la Facultad de Ingeniería*. Tesis de Título. Universidad Tecnológica del Perú. Obtenido de [https://pis1.wikispaces.com/file/view/Presentacion+Final\\_Tesis+I.pdf](https://pis1.wikispaces.com/file/view/Presentacion+Final_Tesis+I.pdf)
- Anillo y Pájaro (2014). *Sistema de información para la gestión de la membresía de iglesias cristianas en Cartagena*. Tesis de Título. Universidad de Cartagena. Obtenido de [http://190.242.62.234:8080/jspui/bitstream/11227/737/1/Tesis\\_Anillo\\_Pajaro.pdf](http://190.242.62.234:8080/jspui/bitstream/11227/737/1/Tesis_Anillo_Pajaro.pdf)
- Benítez (2012). *Introducción de Sublime Text*. Obtenido de <http://www.tecnopedia.net/software/sublime-text-editor-de-texto-para-desarrolladores/>
- Diaz (2007). *Sistema de análisis, diseño e implementación para la administración de comunidades cristianas*. Tesis de Título. Pontificia Universidad Católica del Perú. Obtenido de <http://tesis.pucp.edu.pe/repositorio/handle/123456789/360>
- Iglesia Hoy (2001). *Software desarrollado, Administración de Membresías, finanzas, Escuela Bíblica entre otros*. Obtenido de <http://www.iglesiahoy.com/ihoy/contenido.cfm?cont=MAIN1>
- Ivan, O. M. (2013). *¿QUE ES WAMP SERVER?* Obtenido de *¿QUE ES WAMP SERVER?:* <https://abdonflores.wordpress.com/2013/11/19/que-es-wamp-server/>
- Jorge (2012). *Libro completo de HTML*. Obtenido de <http://es.tldp.org/Manuales-LuCAS/doc-curso-html/doc-curso-html.pdf>
- Javier (2009). *CSS avanzado*. Obtenido de [http://www.jesusda.com/docs/ebooks/css\\_avanzado.pdf](http://www.jesusda.com/docs/ebooks/css_avanzado.pdf)

- Manuel, T. G. (2012). *Metodología Scrum*. Obtenido de Metodología Scrum:  
<http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>
- Lujan (2001). *Programación en internet: clientes web. (Libro completo gratuito en pdf)*. Obtenido de <http://rua.ua.es/dspace/handle/10045/16994>
- Olivo (2014). *Sistema web para administración de la iglesia cristiana bautista SINAI de Ibarra. Colombia*. Obtenido de  
<http://repositorio.utn.edu.ec/bitstream/123456789/4635/1/04%20ISC%20306%20TESIS.pdf>
- Pila (2013). *Desarrollo de un sistema de administración de las actividades eclesiásticas de la iglesia católica parroquial Santiago apóstol usando herramientas web*. Tesis de Título. Escuela Politécnica Nacional de Quito, Ecuador. Obtenido de  
<http://bibdigital.epn.edu.ec/bitstream/15000/7063/1/CD-5248.pdf>
- Raya J., Raya L. y Zurdo J. (2014). *Sistema Informático*.  
Obtenido de <https://es.scribd.com/doc/299531630/Sistemas-Informaticos-Ed-RAMA-2011>
- Rouse (2016). *Gestión de Servicios de TI*. Obtenido de  
<http://searchdatacenter.techtarget.com/es/definicion/ITSM-gestion-de-servicios-de-TI>
- Tovar (2014). *Sistema de Información para la gestión de la membresía de las iglesias cristianas. Tesis de título, Universidad de Cartagena*. Obtenido de  
[http://190.242.62.234:8080/jspui/bitstream/11227/737/1/Tesis\\_Anillo\\_Pajaro.pdf](http://190.242.62.234:8080/jspui/bitstream/11227/737/1/Tesis_Anillo_Pajaro.pdf)

## **Agradecimiento**

A Dios, por cada día de vida que me regala en este mundo, por protegerme y cuidarme siempre y por la bendición de cumplir este objetivo tan anhelado. A mi madre Carmen Mendoza, que con esfuerzo y dedicación me han apoyado en el camino de la vida. Por enseñarme a valorar las pequeñas cosas. Con amor supo guiarme, pero a la vez dejarme tomar mis propias decisiones, y aunque muchas veces me equivoqué y tropecé, me ayudo a levantarme y a seguir adelante. Gracias madre te adoro. A mi hermana, compañera de vida y de travesuras. Me siento afortunado y bendecido por tenerla en mi vida y por estar a mi lado apoyándome. A mi abuelita querida, por facilitarme un rinconcito en el que puedo refugiarme y por preocuparse por mí. A mis amigos queridos, gracias por su amistad y consejos. Pero por sobre todo por apoyarme y quererme tal y como soy. Son un tesoro para mí y siempre los llevaré en mi corazón. Los quiero mucho. A mi director, Ing. Víctor Albinagorta por las enseñanzas, tiempo y apoyo brindados durante mi vida estudiantil y en el desarrollo de mi proyecto de titulación. A todos aquellos que creyeron en mí y que de alguna manera me inspiraron a continuar y a luchar por mis sueños. Este es uno de ellos.

Verdeguer Renzo