



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



Diseño, Desarrollo e Implementación de un Sistema para Administración de Iglesias

Rosa Antonia Coello Ojeda ⁽¹⁾, Alexandra Isabel Cacao Ponce ⁽²⁾,
Ing. Carlos Martín Barreiro, Profesor de la Materia, ESPOL ⁽³⁾,
Facultad de Ingeniería Eléctrica y Computación ⁽¹⁾⁽²⁾⁽³⁾
Escuela Superior Politécnica del Litoral (ESPOL) ⁽¹⁾⁽²⁾⁽³⁾
Campus Gustavo Galindo, Km 30.5 vía Perimetral
Apartado 09-01-5863. Guayaquil-Ecuador
rousicoello@gmail.com ⁽¹⁾, alexita.ahome@gmail.com ⁽²⁾, cmmartin@espol.edu.ec ⁽³⁾

Resumen

En este documento se presentará la propuesta de una aplicación Web para la Administración de Iglesias. Se tratarán diversos aspectos de la aplicación como las bases sobre la cual parte la propuesta realizada, su diseño, desarrollo y los resultados obtenidos durante las diferentes pruebas realizadas antes y después de su implementación, se hablará acerca de como las Iglesias llevan actualmente sus procesos, también se realizará el análisis del mercado ecuatoriano e internacional en cuanto al uso de herramientas que brindan servicios similares al de nuestra aplicación web. Se analizará la solución del problema y su alcance, se definirán los requerimientos funcionales y no funcionales del sistema, así como los usuarios que usaran la aplicación web. Se explicarán las características del sistema: su arquitectura, módulos que tendrá el sistema y diagrama de la base de datos. Al final, luego de obtener los resultados de la implementación del sistema y los resultados de las pruebas se expondrán las conclusiones y se realizaran recomendaciones para la mejora del sistema.

Palabras Claves: *Sistemas, Administración, Software, Iglesias, Aplicaciones web.*

Abstract

In this document we will present the offer of a Web application for the Administration of Churches. Diverse aspects of the application will treat each other as the bases on which it divides the realized offer, its design, development and the results obtained during the different tests we made before and after its implementation, we will explain how the Churches make their processes nowadays, also we will be analyzing the Ecuadorian and international market as the use of tools that offer services similar to the ones in our web application. There will be analyzed the solution of the problem and its scope, there will be defined the functional and not functional requirements of the system, as well as the users who will be using our web application. The characteristics of the system will be explained: his architecture, modules of the system and the database diagram. Finally, after obtaining the results of the implementation of the system and the results of the tests we will expose the conclusions and recommendations for the improvement of the system.

1. Introducción

En la actualidad la mayoría de las iglesias evangélicas manejan sus operaciones de una manera poco eficiente ya que todos sus registros los llevan en papel, lo cual requiere que las personas estén constantemente trabajando en la revisión de documentos. Dentro de una iglesia se lleva control de miembros, inventario, ingresos y egresos, reuniones y grupos, entre otros.

En el Mercado Internacional se han desarrollado diferentes Aplicaciones para Administración de Iglesias pero en el Ecuador no se han desarrollado Sistemas Administradores de Iglesias, sin embargo, algunas iglesias tienen implementado algún sistema desarrollado en otro país.

El Sistema de Administración de Iglesias (SAI) contribuirá a las iglesias a una mejor administración de todos sus recursos, ayudará a mejorar el tiempo de respuesta de los procesos realizados dentro de la iglesia y mantendrá de manera organizada toda la información de interés.

SAI cuenta con los módulos necesarios para llevar un buen control y una administración completa en las Iglesias y es un Sistema capaz de competir con los Sistemas que ya han sido desarrollados en otros países.

Entre otras cosas SAI contará con capacidad para registrar información de Miembros, Usuarios, Grupos, Tesorería, Inventario y Actividades de la iglesia y luego poder ser consultada y modificada.

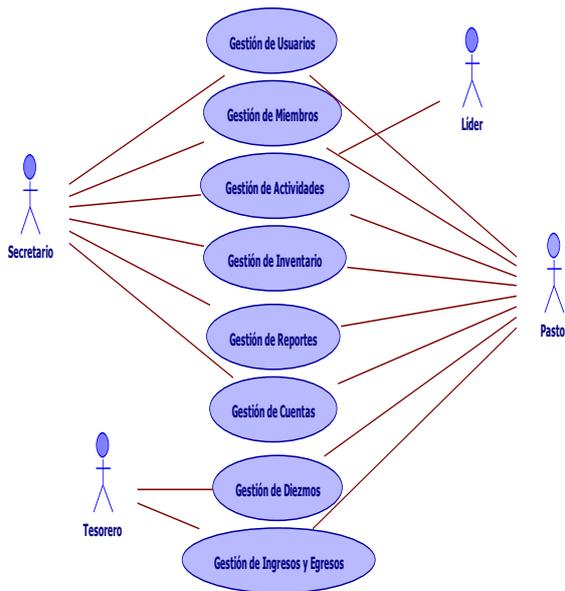


Figura 1 Casos de Uso de SAI (ALTO NIVEL)

2. Diseño del Sistema

El sistema fue implementado bajo una arquitectura de tres capas como se muestra en la figura 1.

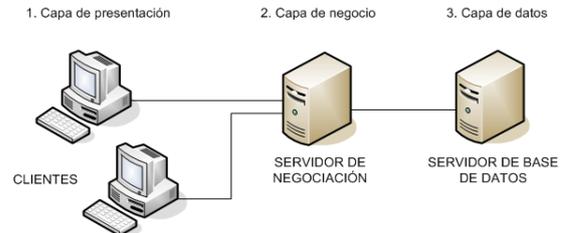


Figura 2 Modelo del tres capas

La capa de presentación está diseñada para ofrecer al cliente un entorno amigable y fácil de utilizar mostrando lo indispensable para un buen manejo del sistema desde un menú desplegable que indica explícitamente lo que se puede realizar. Fue desarrollada con lenguaje asp.net, html, javascript y css.

La capa de negocios se encarga de recibir todos los requerimientos del usuario al momento de interactuar con el sistema y ejecuta toda la lógica necesaria utilizando instrucciones escritas en lenguaje c#.

La capa de datos está implementada con una base de datos en Microsoft SQL Server 2008 ya que es la tecnología más compatible con .Net y facilita las operaciones necesarias entre el sistema y la base de datos. Se utilizó el Entity Data Model de ADO .Net para el tratamiento de la información referente a usuarios, grupos, inventario, tesorería y demás..

Para el control de acceso al sistema y la seguridad se lleva registro de usuarios y contraseñas.

En el sistema los contenidos son filtrados dependiendo del rol que ocupa en la iglesia la persona que ingresa al mismo.

Para la autenticación de los usuarios al sistema utilizamos de manera transparente un procedimiento almacenado de la base de datos, el cual nos entrega como resultado todos los datos del usuario en caso de haber ingresado correctamente los datos con lo que desde la lógica de negocio podemos conocer fácilmente si el usuario es válido o no.

2.1 Requerimientos de hardware

La aplicación será utilizada en la intranet de la Iglesia por lo que se necesita un servidor web que soporte tantos usuarios como roles existan en la iglesia para que de ser ideal cada uno pueda trabajar en su propia área al mismo tiempo.



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



Se podría tener al Servidor de Bases de datos en el mismo computador por lo que se debería tener un sistema de respaldo como por ejemplo RAID para proteger la integridad y la tolerancia a fallos del sistema.

Se debe establecer una conexión de red con las demás computadoras de la iglesia que tendrán acceso a la intranet y en el servidor se deberá configurar un nombre de dominio fácil de recordar para los usuarios del sistema.

2.2 Requerimientos de software

En cuanto a software del servidor, deberá soportar ASP. Lo natural para ejecutar páginas web en ASP es utilizar el servidor IIS (Internet Information Server) de Microsoft, la empresa encargada de la tecnología ASP y para esto se debería tener un Windows server.

En el caso de las estaciones de trabajo finales lo único que se requiere es un navegador web que soporte los estándares HTML y XHTML.

3. Implementación del Sistema

A continuación describiremos las tecnologías utilizadas para la implementación del sistema.

Se utilizó *ASP.Net* porque tiene un buen nivel de abstracción en la construcción de sitios web ya que se pueden crear rápidamente basándose en los controles incluidos en el *framework*.

Por otro lado tenemos *C#.Net* porque es uno de los lenguajes de programación compatibles con *ASP.Net* y nos permite realizar operaciones robustas y completas para la definición del modo en que debe funcionar la aplicación.

Tanto el *Ajax Control Toolkit* y *Java Script* fueron utilizados al momento de desarrollar la capa de presentación de SAI porque permiten incorporar en nuestra página efectos, animaciones y eventos del lado del cliente sin tener que hacer consumo de recursos del lado del servidor así como las ventajas propias de *Ajax* que aumentan la interactividad, velocidad y usabilidad en las aplicaciones.

En todos los formularios para ingresos disponibles en el sistema se utiliza una herramienta de *Ajax Control Toolkit*, se trata del *RequiredFieldValidator* y del *ValidatorCalloutExtender* los cuales permiten la validación de campos requeridos del formulario que el usuario esté dejando sin completar y le muestra un mensaje informándole que son obligatorios.

3.1 Herramientas del lado del cliente

En algunas partes del proyecto se utilizaron librerías como jQuery que es un framework de

JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas web. También se utilizó FullCalendar que es un plugin jQuery que provee un calendario drag & drop. Es de código abierto.

3.2 Herramientas del lado del servidor

Otra de las herramientas que se utilizaron para el desarrollo de la aplicación están ADO.NET es un conjunto de componentes de software, una parte de la biblioteca de clases base que están incluidas en el Microsoft .NET Framework que hemos utilizado para acceder y para modificar los datos almacenados en nuestro Sistema Gestor de Bases de Datos Relacionales, aunque también puede ser usado para acceder a datos en fuentes no relacionales.

Entity Data Model (EDM) es un conjunto de conceptos que describen la estructura de los datos, independientemente del formato en el que estén almacenados. EDM se basa en el modelo entidad-relación, pero también incorpora nuevas funciones y amplía sus usos tradicionales.

Crystal Reports es una aplicación de inteligencia empresarial utilizada para diseñar y generar informes desde una amplia gama de fuentes de datos.

3.4 Implementación del sistema

En SAI la representación de la Capa de Presentación esta en Formularios, Informes y Respuestas a Usuarios usando *ASPX.net*, *HTML*, *Java Script* y así mostrar al usuario la información del Sistema. Se utiliza una *masterpage* para lograr un esquema fijo en todas las páginas del sistema.

Para la capa de negocio se utilizó el lenguaje *C#.Net*. En general se hacen las debidas validaciones de menús y de páginas por roles según la autenticación de un usuario.

Se hacen inserciones y consultas utilizando al *Entity Data Model*, instanciando objetos para la posterior utilización que nos permita traer los datos desde la base ejecutando procedimientos almacenados o simples queries.

Para la implementación de la capa de datos utilizamos como motor de base de datos a Microsoft *SQL Server 2008* y el *Entity Data Model* de *ADO .Net*.

3.5 Módulos de SAI

SAI ayuda a llevar un Control de diferentes procesos que se realizan dentro de las iglesias. Para esto SAI está dividida en cinco módulos.



Figura 3 Página inicial de SAI

3.5.1 Módulo Sistema

En este módulo se lleva control de los Miembros, Grupos y Cargos que existen en la Iglesia, así como de los Usuarios que ingresan al Sistema.

Para el Control de los Miembros de la iglesia es necesario el Ingreso de información personal en general e información cristiana. Esto se logra con la implementación de formularios mediante un entorno fácil de utilizar que interactúa en todo momento con el usuario del sistema indicándole la manera correcta en la que debe completar los registros. Este formulario de ingreso también permite cargar la foto del miembro de la iglesia para tener información más completa a la hora de consultarla o editarla.

Un nuevo usuario del sistema puede ser agregado dentro del formulario para ingreso de Miembro al momento de registrar un nuevo miembro o después desde el menú. Para poder ser un usuario del sistema es necesario ser un Miembro de la Iglesia y los permisos que tenga este usuario dentro del sistema dependerán de su cargo dentro de la iglesia.

El usuario administrador del sistema tiene la opción de crear nuevos cargos y nuevos grupos de la iglesia de manera que sería preferible que la persona encargada de este usuario tenga conocimientos suficientes tanto de lo que ocurre en la iglesia como del uso de la tecnología.

3.5.2 Módulo Tesorería

Este módulo fue creado para llevar control del dinero que ingresa y egresa de la iglesia. Existen dos tipos de ingresos y egresos, los generales y los de grupos. Tanto los generales como los de grupos son para ingresos y egresos de cualquier clase y el sistema permite especificar el origen y motivo del ingreso o egreso al momento de registrarlo, consultarlo o modificarlo.

Al tener registro de toda esta información el sistema permite generar reportes de todo tipo para que las autoridades tengan un buen control de la economía de la iglesia.

3.5.3 Módulo Inventario

Este módulo fue diseñado para llevar el control de todos los bienes que le pertenecen a la iglesia. El sistema permite registrar cada vez que un ítem ingresa o egresa de las instalaciones de la iglesia especificando si se trata de compras o donaciones en el caso que sea la primera vez, también se puede consultar y modificar la información así como se puede dar de baja algún ítem que este dentro de la iglesia. La información sobre el inventario puede ser mostrada mediante reportes generados a conveniencia de quien lo requiera.

3.5.4 Módulo Actividades

Este módulo fue creado para llevar el control de reuniones, cultos, actividades organizadas por la iglesia, entre otras. Solamente existe un usuario en el sistema que puede crear o editar actividades generales de la iglesia, el secretario y el pastor. Los demás usuarios solamente pueden consultar estas actividades en el calendario. Dicho calendario muestra las actividades pertinentes a cada usuario, es decir que si por ejemplo el usuario es un miembro del grupo de damas, entonces las actividades mostradas serán solamente las generales de la iglesia y las de su propio grupo. A diferencia del calendario del pastor que muestra todas las actividades generales y de grupos.

3.5.5 Módulo de Reportes

Del modulo de Tesorería e Inventario se tiene mucha información importantes para la parte de reportes, el modulo de reportes recoge los datos de estos dos módulos para generar reportes a las autoridades de la Iglesia.

4. Conclusiones

Algunos de los objetivos que nos planteamos al principio del proyecto fueron principalmente: analizar la manera en que se administra una iglesia en la actualidad, analizar diferentes herramientas nuevas de desarrollo web, diseñar un sistema que permita a una iglesia almacenar información pertinente para su administración.

Estos objetivos se cumplieron en su totalidad y podemos concluir que:

1. Una de las falencias que tiene la administración de la mayoría de iglesias en



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



- Ecuador es que actualmente toda la información se encuentra almacenada en papel lo cual no es productivo ya que el papel es un material que podría deteriorarse o perderse y resulta tedioso manejarlo al momento de intentar agregar u obtener información sobre la iglesia.
2. No existen muchos sistemas para administrar una iglesia en la actualidad. En su mayoría son aplicaciones de escritorio difícilmente portables que no cuentan con herramientas modernas, rápidas y de fácil uso. Por lo tanto SAI quiso ofrecer todo esto y más en una sola aplicación web que podrá llegar a más lugares y usuarios por ser multiplataforma.
 3. Para las iglesias nuestro sistema les ayudará a tener una mejor organización de su información; ayudará a que todo se lleve de una manera más transparente y permitirá a todos sus miembros involucrarse un poco más con la administración de la misma.
 4. Para cumplir con todos los objetivos de SAI se establecieron los requerimientos funcionales y no funcionales del sistema considerando características observadas en las herramientas de mercado y en los resultados obtenidos en el sondeo de opinión que se realizó a los miembros de una iglesia evangélica local. Los miembros de la iglesia indicaron que las principales ventajas del uso de una herramienta para la administración es que esta sería mucho más fácil y amigable, la información puede ser almacenada de manera segura y puede ser consultada rápidamente.
 5. A la vez se analizó las necesidades o aspectos que los miembros de la iglesia consideran importantes en un sistema para administración y la mayoría coincidió en que el uso de la tecnología ayudará a un crecimiento intelectual dentro de la iglesia y la administración será más eficiente.
 6. Hablando del diseño de la aplicación, fue una buena decisión utilizar el Entity Data Model como medio para crear un modelo conceptual a partir de la base de datos diseñada para el sistema. Esto hace que el tiempo de respuesta del sistema sea menor y que los datos se almacenen y modifiquen de manera segura.

5. Recomendaciones

Las recomendaciones que damos para nuevas versiones del sistema son las siguientes:

1. Permitir que la aplicación web sea accedida desde el exterior incorporando los debidos

permisos para mantener la seguridad e integridad de la información.

2. Creación dinámica de documentos como cartas Pastorales para los miembros entre otros documentos tomando como referencia la información que es almacenada en el Sistema.
3. Incorporar la parte de Ministerios de los miembros de la Iglesia, los ministerios son otro tipo de cargo que tienen los Miembros en la iglesia.
4. Creación de Roles de Pago para la parte de Recursos Humanos de la Iglesia.

6. Bibliografía

- [1] José Manuel Sánchez Suarez, Introducción a JQuery, <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=jQueryUI>, Junio 2010.
- [2] John Resig, JQuery, <http://es.wikipedia.org/wiki/JQuery>, Julio 2010.
- [3] Adam Shaw, Full Callendar, <http://arshaw.com/fullcalendar/>, Mayo 2010.
- [4] Microsoft Corporation, Entity Data Model, <http://msdn.microsoft.com/es-es/library/ee382825.aspx>, Agosto 2010.
- [5] Bean Software, Crystal Reports en ASP .Net, <http://www.beansoftware.com/ASP.NET-Tutorials/Using-Crystal-Reports.aspx>, Septiembre 2010.
- [6] Mauro Sant'Anna, Reportes en .NET con Crystal Reports, <http://www.willydev.net/InsiteCreation/v1.0/descargas/prev/crearcystal.pdf>, Septiembre 2010.
- [7] Licencia Creative Commons Atribucion Compartir Igual 3.0, ADO .NET, <http://es.wikipedia.org/wiki/ADO.NET>, Julio 2010.
- [8] Msc Carlos Martin, Conocimientos AJAX con ASPx, Seminario de graduación Desarrollo de aplicaciones Web avanzadas con Ajax, Noviembre 2009 – Mayo 2010.