



INSTITUTO TECNOLÓGICO
“CORDILLERA”

CARRERA DE ANÁLISIS DE SISTEMAS

REGISTRO Y CONTROL DE INFORMACIÓN PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMÁTICO PARA EL COMITÉ
PRO-MEJORAS SANTA ISABEL

Proyecto de investigación previo a la obtención del título de Tecnólogo en Analista
de Sistemas

Autor: Ayala Cuichán Alicia Alexandra

Tutor: Ing. Richard Mafla, Msc.

Quito, Abril 2014



DECLARACIÓN DE APROBACIÓN TUTOR

En mi calidad de tutor del trabajo sobre el tema: **“REGISTRO Y CONTORL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL”**, presentado por la ciudadana: Ayala Cuichan Alicia Alexandra, estudiante de la Escuela de Análisis y Sistemas, considero que dicho informe reúne los requisitos y méritos suficientes para ser sometido a la evaluación por parte del Tribunal de Grado, que el Honorable Consejo de Escuela designe, para su correspondiente estudio y calificación.

Quito, Abril del 2014

Ing. Richard Mafla

TUTOR

Ing. Jhonny Coronel

LECTOR

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL



DECLARATORIA

Declaro que la investigación es absolutamente original, auténtica, personal, que se han citado las fuentes respectivas y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes. Las ideas, doctrinas resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

Ayala Cuichan Alicia Alexandra

C.C. 1721833166

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, el estudiante Ayala Cuichan Alicia Alexandra, por sus propios y personales derechos, a quien en lo posterior se le denominará el "CEDENTE"; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el "CESIONARIO". Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

PRIMERA: ANTECEDENTE.- a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado "Registro y Control de información personal y pago de alcúotas mediante un sistema informático para el Comité Pro- Mejoras Santa Isabel", el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. b) Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



SEGUNDA: CESIÓN Y TRANSFERENCIA.- Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita todos los derechos patrimoniales del programa de ordenador descrito en la cláusula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.). El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

TERCERA: OBLIGACIÓN DEL CEDENTE.- El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



CUARTA: CUANTIA.- La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

QUINTA: PLAZO.- La vigencia del presente contrato es indefinida.

SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.- Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se solicitará la asistencia de un Mediador del Centro de Arbitraje y Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas: a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvención, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

SÉPTIMA: ACEPTACIÓN.- Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 21 días del mes de abril del dos mil catorce.

f) _____

C.C. 1721833166

CEDENTE

f) _____

Instituto Superior Tecnológico Cordillera

CESIONARIO



AGRADECIMIENTO

Primero y antes que nada, dar gracias a mi Dios, por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a las personas que han sido mi soporte y compañía durante todo el periodo de estudio.

Agradecer hoy y siempre a mi familia por el esfuerzo realizado por ellos. El apoyo en mis estudios, de ser así no hubiese sido posible. A mis padres y demás familiares ya que me brindan el apoyo, la alegría y me dan la fortaleza necesaria para seguir adelante.

Un agradecimiento a mi asesor el Ingeniero Richard Mafla por todo el apoyo brindado en todo momento.

A mis amigos y compañeros que fueron parte de mi vida en el instituto y estuvieron involucrados en cada experiencia, gracias por estar conmigo siempre y dejarme aprender de ustedes.

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



DEDICATORIA

A mi Dios ya que es él quien me ha dado fortaleza, sabiduría y constancia para seguir adelante, A mis Padres Silvia y Fabián que son los principales gestores de este logro, a mi hermano Anthony que siempre está junto a mí, a todos mis familiares y mis amigos que día a día me están apoyando y dando ánimos para seguir adelante.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



INDICE GENERAL

Contenido

DECLARACIÓN DE APROBACIÓN TUTOR	i
DECLARATORIA	ii
CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL.....	iii
AGRADECIMIENTO	vii
DEDICATORIA	viii
RESUMEN EJECUTIVO	xvi
ABSTRACT	xvii
CAPITULO I: Antecedentes	1
1.01 Contexto	1
1.02 Justificación	2
1.03 Definición del Problema	3
1.03.01 Matriz de Fuerzas	3
CAPITULO II: ANÁLISIS DE INVOLUCRADOS	5
2.01 Mapeo De Involucrados.....	5
2.02 Matriz de Análisis de Involucrados	6
CAPITULO III: PROBLEMAS Y OBJETIVOS	7
3.01 Árbol de Problemas	7
3.02 Árbol de Objetivos	9
CAPITULO IV: ANÁLISIS DE ALTERNATIVAS.....	12
4.02 Matriz de Análisis de impacto de los Objetivos.....	14
4.03 Diagrama de Estrategias	16
4.04 Matriz de Marco Lógico	17
CAPÍTULO V: Propuesta	21
Justificación Técnica.....	21
5.02 Análisis y Diseño	23
5.02.01 Diagrama de Caso de Uso General	23



5.02.02	Diagramas de Realización	24
5.02.03	Diagramas de Secuencia	28
5.02.04	Diagrama de Colaboración	31
5.02.05	Diagrama de Implementación.....	36
5.02.06	Diagrama de Clases.....	37
5.02.07	Modelo Lógico	38
5.02.08	Modelo Físico	40
5.03	Desarrollo.....	41
5.03.01	Arquitectura Del Sistema	41
5.03.01.01	Capa de Presentación:	41
5.03.01.02	Capa de Negocio	41
5.03.01.03	Capa de Datos:	41
5.03.02	Módulos.....	42
5.03.02.01	Módulo de Seguridad.....	42
5.03.02.02	Módulo de Mantenimiento	42
6.05.06	Módulo Lógica Negocios.....	43
5.03.03	Estándares	43
5.03.03.01	Estándares de Programación	43
5.03.03.01.01	Declaración de Variables.....	43
5.03.03.01.04	Declaración de variables, atributos y objetos	44
5.03.03.01.06	Declaración de Datos	45
5.03.03.01.07	Declaración de Clases	45
5.03.03.01.08	Declaración de Funciones y Procedimientos.....	46
5.03.03.01.08	Declaración de Métodos.....	46
5.03.03.02	Estándares de Base de Datos.....	47
5.03.03.02.02	Tipo de Datos	47
5.03.03.02.02	Tablas	48
5.03.03.03	Estándares de Diseño	49
5.03.03.03.01	Estándares de Diseño UML.....	49
5.05	Diseño de Interfaces.....	49



5.05	Pruebas de módulo.....	53
5.05.01	Módulo de Seguridad.....	53
5.05.02	Módulo Mantenimiento.....	54
5.05.03	Pruebas de Interfaz de Usuario.	54
5.05.04	Pruebas de Desempeño.	55
5.05.05	Pruebas de Carga.....	56
5.05.06	Pruebas de Validación.....	56
CAPITULO VI: Aspectos Administrativos		57
6.01	Recursos	57
6.02	Presupuesto	58
6.03	Cronograma.....	60
CAPITULO VII: Conclusiones y Recomendaciones.....		61
7.01	Conclusiones.....	61
7.02	Recomendaciones	61
Para que se pueda cumplir los requerimientos del barrio se recomienda:		61
ANEXOS		63
ANEXO 01.....		63
ANEXO 02.....		66
8	Instalación de programas utilizados.....	69
Instalación de Visual Studio 2010		69
INSTALACION DE SQL SERVER 2008.....		80
SCRIPT DE LA BASE DE DATOS.....		100
Diccionario de Datos.....		109
MANUAL TÉCNICO.....		114
Manual de Usuario.....		165
GLOSARIO		175
WEBGRAFIA.....		176



INDICE DE TABLAS

Tabla 1	<i>Tabla de matriz de Fuerzas</i>	3
Tabla 2	<i>Matriz de Análisis de Involucrados</i>	6
Tabla 3	<i>Matriz de Análisis de Alternativas</i>	12
Tabla 4	<i>Matriz de Análisis de Impacto</i>	14
Tabla 5	<i>Caso de Uso de la consulta de valores a pagar</i>	25
Tabla 6	<i>Caso de uso de realizar pago</i>	26
Tabla 7	<i>Caso de uso de generar respuesta</i>	27
Tabla 8	<i>Declaración de variables</i>	44
Tabla 9	<i>Tipo de Controles</i>	44
Tabla 10	<i>Tipo de Datos</i>	45
Tabla 11	<i>Declaración de clases</i>	45
Tabla 12	<i>Nombres de Funciones y Procedimientos</i>	46
Tabla 13	<i>Declaracion de Métodos</i>	46
Tabla 14	<i>Tipo de Datos</i>	47
Tabla 15	<i>Nombres de las tablas de la base de datos</i>	49
Tabla 16	<i>Ingreso al sistema</i>	49
Tabla 17	<i>Ingreso de un Nuevo Socio</i>	50
Tabla 18	<i>Lista de Socios Ingresados</i>	51
Tabla 19	<i>Agregar Clientes</i>	52
Tabla 20	<i>Recursos</i>	57
Tabla 21	<i>Presupuesto</i>	58



INDICE DE FIGURAS

<i>Figura 1:</i> Mapeo de involucrados	5
<i>Figura 2:</i> Árbol de Problemas	8
<i>Figura 3:</i> Árbol de Objetivos	10
<i>Figura 4:</i> Diagrama de Estrategias	17
<i>Figura 5:</i> La figura nos muestra el funcionamiento en el modelado del sistema, siendo este el proceso que se quiere automatizar, para optimizar tanto recursos financieros como humanos.	24
<i>Figura 6:</i> Indica el proceso que se ejecuta cuando se realiza una consulta.	25
<i>Figura 7:</i> Indica el proceso que se ejecuta cuando se realiza el pago	26
<i>Figura 8:</i> Indica el proceso que se realiza cuando se genera una respuesta.....	27
<i>Figura 9:</i> La secuencia de consulta muestra la interfaz del control de acceso validando la información ingresada y mostrando las interfaces correspondientes.	28
<i>Figura 10:</i> La secuencia indica como el socio realiza el pago.	28
<i>Figura 11:</i> Muestra los procesos a seguir para generar el comprobante de pago.	29
<i>Figura 12:</i> Muestra los procesos a seguir para generar el informe.....	29
<i>Figura 13:</i> La secuencia indica el registro de la actividad realizada.	30
<i>Figura 14:</i> Muestra los procesos a seguir para realizar el acta generada en la asamblea.	30
<i>Figura 15:</i> La secuencia indica el proceso que se realiza cuando se genera una respuesta	31
<i>Figura 16:</i> Indica el proceso que se ejecuta cuando se realiza una consulta.	32
<i>Figura 17:</i> La figura muestra el procedimiento para realizar el pago	32
<i>Figura 18:</i> Indica el proceso que se realiza para generar el comprobante de pago	33
<i>Figura 19:</i> Proceso que se realiza para generar un informe de pagos	33
<i>Figura 20:</i> Indica el proceso que se ejecuta cuando se registra una actividad	34
<i>Figura 21:</i> Indica el proceso que se ejecuta cuando se realiza el acta.....	34
<i>Figura 22:</i> Proceso que se realiza cuando se genera una respuesta.	35
<i>Figura 23:</i> Se detalla los componentes de software necesario para la ejecución de la aplicación.	36
<i>Figura 24:</i> La figura muestra un diagrama que recopila toda la información necesaria para el diagrama físico de flujo de datos, este esquema es la base principal para el modelamiento y creación de la base de datos.....	38
<i>Figura 25:</i> La figura muestra el modelo lógico del sistema, en donde se encuentran las tablas correspondientes a la base de datos la cual almacena los datos para la utilización de la aplicación.	39
<i>Figura 26:</i> La figura representa el modelamiento de la base de datos.	40
<i>Figura 27:</i> Ingreso al sistema.....	50
<i>Figura 28:</i> Ingreso de nuevos socios.....	51
<i>Figura 29:</i> Listado de los socios registrados.	52



<i>Figura 30: Ingreso de nueva acta.....</i>	<i>53</i>
<i>Figura 31: Cronograma</i>	<i>60</i>
<i>Figura 32: Ejecutar el instalador</i>	<i>69</i>
<i>Figura 33: Iniciar la instalación</i>	<i>70</i>
<i>Figura 34: Primer paso de instalación.</i>	<i>71</i>
<i>Figura 35: Aceptación de términos de Licencia</i>	<i>72</i>
<i>Figura 36: Seleccionar las características a instalarse</i>	<i>73</i>
<i>Figura 37: Seleccionar lenguajes para instalar</i>	<i>74</i>
<i>Figura 38: Espera de Instalación</i>	<i>75</i>
<i>Figura 39: Finalización de instalación</i>	<i>75</i>
<i>Figura 40: Reinicio del computador después de la instalación</i>	<i>76</i>
<i>Figura 41: Iniciar Visual Studio.....</i>	<i>77</i>
<i>Figura 42: Elegir configuración de entorno.</i>	<i>78</i>
<i>Figura 43: Finalización de la instalación</i>	<i>79</i>
<i>Figura 44: Esta es la primera ventana que se presenta cuando se ejecuta el instalador</i>	<i>80</i>
<i>Figura 45: Aquí se verifica todas las reglas necesarias para la instalacion.....</i>	<i>81</i>
<i>Figura 46: Damos clic en instalación para que la instalación empiece.....</i>	<i>82</i>
<i>Figura 47: En esta ventana solo damos clic en siguiente ya que por defecto viene la clave del producto.....</i>	<i>83</i>
<i>Figura 48: Aquí aceptamos los términos de licencia.</i>	<i>84</i>
<i>Figura 49: Aquí seleccionamos las características que se agregaran a la intalacion.</i>	<i>85</i>
<i>Figura 50: aquí configuramos la instancia, el ID y la dirección donde se realizara la instalación.</i>	<i>86</i>
<i>Figura 51: Aquí nos verifica el espacio en el disco</i>	<i>87</i>
<i>Figura 52: En esta ventana configuramos el servidor.....</i>	<i>88</i>
<i>Figura 53: Damos clic en siguiente.</i>	<i>89</i>
<i>Figura 54: Figura Motor de Base de Datos.....</i>	<i>90</i>
<i>Figura 55: Figura Analysis Services</i>	<i>91</i>
<i>Figura 56: Aquí configuramos el motor de la base de datos</i>	<i>92</i>
<i>Figura 57: Configuración de Analisis Services.....</i>	<i>93</i>
<i>Figura 58: Configuración de Reporting Services.....</i>	<i>94</i>
<i>Figura 59: Esta ventana nos indicara un informe de los errores encontrados.....</i>	<i>95</i>
<i>Figura 60: En esta ventana nos mostrara las reglas necesarias para la instalación.</i>	<i>96</i>
<i>Figura 61: Esta ventana nos indica que estamos listos para empezar la instalación.</i>	<i>97</i>
<i>Figura 62: Esta ventana nos muestra el progreso de la instalación.</i>	<i>97</i>
<i>Figura 63: Progreso de la instalación.....</i>	<i>98</i>
<i>Figura 64: Esta ventana nos indica que la instalación se realizó correctamente.</i>	<i>98</i>
<i>Figura 65: Aquí ya está instalado el programa y listo para usarse</i>	<i>99</i>
<i>Figura 66: Diccionario de datos de la tabla TBL_DFACTURA</i>	<i>110</i>



<i>Figura 67:</i> Diccionario de datos de la tabla TBL_DOCUMENTOSENTREGADOS	110
<i>Figura 68:</i> Diccionario de datos de la tabla TBL_DOCUMENTOSRECIBIDOS	111
<i>Figura 69:</i> Diccionario de datos de la tabla TBL_FACTURA.....	111
<i>Figura 70:</i> Diccionario de datos de la tabla TBL_HISTORIAL.....	112
<i>Figura 71:</i> Diccionario de datos de la tabla TBL_LOTE.....	112
<i>Figura 72:</i> Diccionario de datos de la tabla TBL_ROL	112
<i>Figura 73:</i> Diccionario de datos de la tabla TBL_SOCIO.....	113
<i>Figura 74:</i> Diccionario de datos de la tabla TBL_TIPO	113
<i>Figura 75:</i> Diccionario de datos de la tabla TBL_TIPODOCUMENTO.....	113
<i>Figura 76:</i> Diccionario de datos de la tabla TBL_USUARIO.....	114
<i>Figura 77:</i> Ingreso al sistema	165
<i>Figura 78:</i> Ventana principal de acceso al sistema	166
<i>Figura 79:</i> Ingreso de nuevo socio	167
<i>Figura 80:</i> Listado de socios registrados.....	168
<i>Figura 81:</i> En esta ventana podremos realizar actas	169
<i>Figura 82:</i> Aquí encontramos el listado de las actas registradas.....	170
<i>Figura 83:</i> En esta ventana realizamos el cuerpo del acta.	171
<i>Figura 84:</i> Documento creado.	172
<i>Figura 85:</i> Ingreso de nuevos Usuarios.....	173
<i>Figura 86:</i> Aquí encontramos el listado de los usuarios registrados	174



RESUMEN EJECUTIVO

El presente proyecto de tesis propone la implementación de un Software de Registro y control de pago de alcúotas para el Comité Pro-mejoras Santa Isabel de la Ciudad de Quito. Esta propuesta se da debido al control inadecuado que se lleva dentro de la organización.

Con la falta de herramientas tecnológicas ha aumentado el índice de pérdida de documentos, duplicidad de funciones laborales y pagos irreales por parte de los socios.

El barrio necesita automatizar los procesos financieros ya que estos no se realizan con efectividad y agilidad. En este trabajo se presenta el desarrollo de módulos de reportes, en donde podremos imprimir los reportes de los pagos generados; un módulo administrativo el cual nos permitirá manejar el proceso de factura, cuotas y multas.

Con los avances tecnológicos que hoy en día se están realizando las organizaciones pueden poseer tecnología de calidad, debido a esto el barrio al ver las necesidades que la misma posee desea implementar tecnología la cual ayudará a controlar los procesos y procedimientos que internamente se realizan.

La principal ventaja de esta aplicación es que la directiva del barrio puede tener una mejor toma de decisiones en el ámbito de cuentas.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



ABSTRACT

This thesis project proposes the implementation of a Software Registration and payment control aliquots for the Pro - Santa Isabel Improvements Committee of the City of Quito. This proposal is given due to inadequate control carried within the organization .

With the lack of technological tools has increased the rate of loss of documents, duplication of job functions and unrealistic payments by partners .

The district needs to automate financial processes as these are not performed effectively and agility. In this paper the development of reporting modules , where we can print reports of payments generated is presented ; an administrative module which will allow us to manage the process of billing , fees and fines.

With advances in technology today are being made organizations can have quality technology because this neighborhood needs to see that it has want to deploy technology which will help to monitor the processes and procedures that are performed internally .

The main advantage of this application is that the policy of the district may have a better decision making in the field of accounts.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



CAPITULO I: Antecedentes

1.01 Contexto

En el cantón Quito, Parroquia Cotacollao los moradores del Comité Pro-mejoras Santa Isabel aprovechando el beneficio que la naturaleza brindaba con el afloramiento de las vertientes de Mindo y en vista de la necesidad del líquido vital se organizaron e iniciaron los trabajos para obtener el líquido vital, los mismos que a través de mingas han logrado obtener el servicio de agua de riego.

El barrio al ser una entidad autónoma todas sus actividades lo sustentan con recursos propios, los mismos que son recaudados por los servicios de agua de riego y las instalaciones de nuevas acometidas.

Actualmente la organización no cuenta con una adecuada información financiera necesaria, que permita conocer con razonabilidad los saldos de las cuentas de los diferentes socios presentados a una fecha determinada, debido a las malas prácticas administrativas, financieras y un control deficiente.

El sistema de registro y control de pagos de alícuotas es una herramienta muy importante, para proporcionar seguridad razonable de que se están cumpliendo con la eficiencia eficacia y economía de las operaciones y calidad en los servicios que prestan. Por lo tanto es indispensable la aplicación adecuada de un control en todas las áreas de las organizaciones y demás instituciones.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



1.02 Justificación

Al ver la importancia que tiene el Registro y Control para apoyar a la administración financiera, es un tema de impacto; puesto que el entorno político, económico y social genera cambios en las organizaciones sociales, los riesgos varían, las necesidades son otras, las culturas se transforman, es por tal motivo que la organización se ve obligada a fortalecer la administración financiera y mejorar el servicio a la comunidad.

Estimular la instauración de la visión de la organización, ya que es un organismo sin fines de lucro, de derecho privado, autónomo e independiente, teniendo como esencia el de otorgar y controlar las concesiones del servicio y mantenimiento del líquido vital.

Como miembro del barrio contando con los conocimientos adquiridos en el Instituto y al palpar la realidad de la organización es de vital importancia aportar con esta investigación para apoyar en la buena administración y el buen uso de los fondos económicos de la misma.

En la actualidad el registro y control de las mingas, reuniones y pagos se lo hace manualmente teniendo como consecuencia una mala información financiera, debido a que muchas veces los datos son irreales.



1.03 Definición del Problema

1.03.01 Matriz de Fuerzas

La matriz T identifica las fuerzas que impulsan y bloquean la solución del problema central o global relacionadas con el proyecto.

Tabla 1

Tabla de matriz de Fuerzas

PROBLEMÁTICA AGRAVADA	SITUACIÓN ACTUAL				SITUACIÓN MEJORADA		
Ausencia de procesos de registro y control de pagos de alícuotas	No cuenta con una adecuada información financiera				Mejor registro y control de pagos de alícuotas		
Fuerzas Impulsadoras	I	P	C	I	P	C	Fuerzas Bloqueadoras
Contar con los directivos del barrio para la ejecución del proyecto	3	4	4	3	No contar con los directivos del barrio		
Dejar atrás la forma manual en la que se realiza el registro de los pagos.	4	4	4	5	No tener con el presupuesto necesario para realizar el proyecto.		
Proporcionar a los socios una mejor información financiera	4	3	2	4	No contar con la información necesaria para realizar el software		
Tener un mejor control de los socios que asisten a mingas y reuniones.	4	3	2	3	La persona a manejar el sistema no tenga un buen conocimiento sobre informática		
Dar una mejor presentación al barrio.	2	3					
Emitir informes concretos y oportunos que permita a los directivos tomar decisiones oportunas e inmediatas	4	4					

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL



Nota:

PC Potencial de Cambio

I: Intensidad

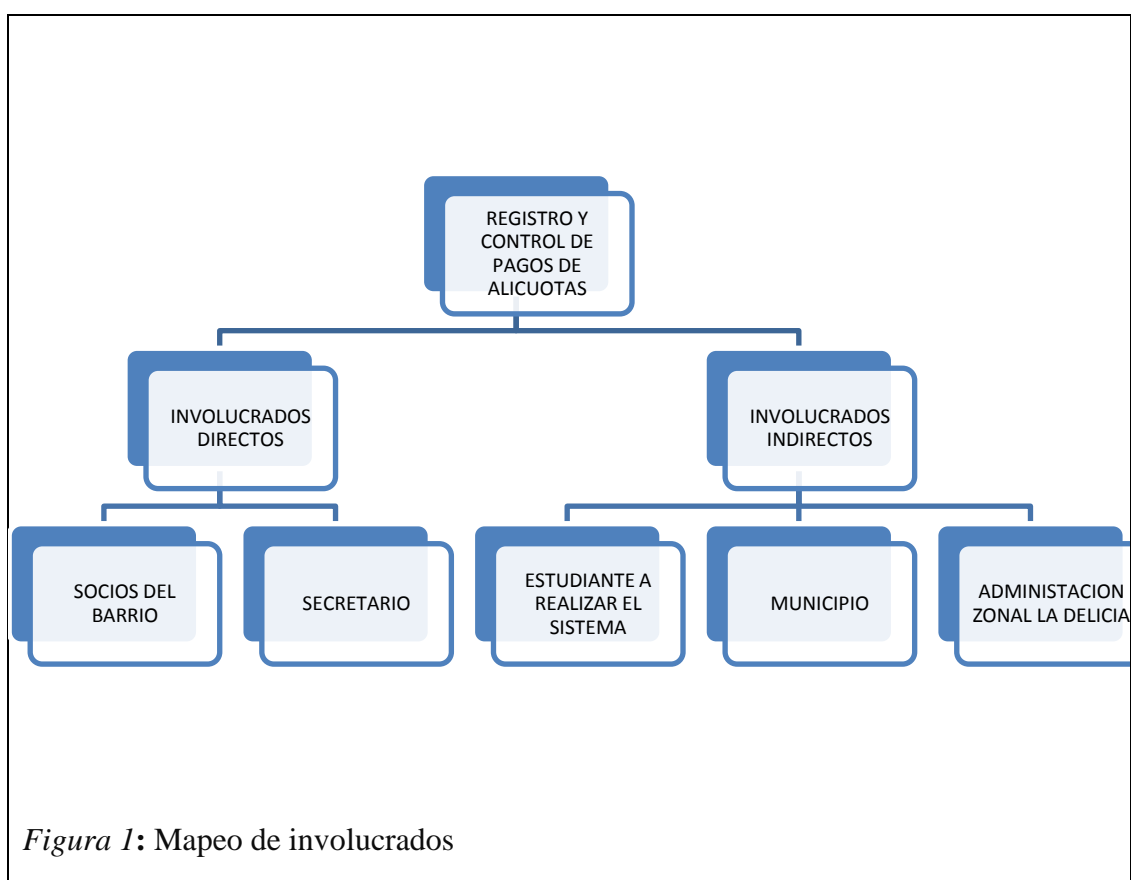
La matriz T nos indica que el barrio tiene ausencia de procesos de registro y control de pagos de alcúotas obteniendo así una inadecuada información financiera para ello las fuerzas impulsadoras es tener un mejor control de los socios que asisten a mingas y reuniones, dar una mejor presentación y emitir informes concretos que permita a los directivos tomar decisiones oportunas e inmediatas y así mejorar este proceso.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***

CAPITULO II: ANÁLISIS DE INVOLUCRADOS

2.01 Mapeo De Involucrados

Esta técnica trata de identificar a los grupos que de manera: directa o indirecta están relacionados con el problema que se desea solucionar o las situaciones que quieren ser modificadas.



Tenemos como involucrados a los socios del barrio y al secretario ya que son las personas que van a actuar directamente con el software y los involucrados indirectos al estudiante a realizar el software, municipio y la administración zonal La Delicia ya que de una u otra manera interactúan con el barrio.

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL



2.02 Matriz de Análisis de Involucrados

La matriz de análisis de involucrados nos ayuda a identificar a las personas y organizaciones interesadas en el proyecto, así como también en las políticas, los recursos, y las responsabilidades institucionales, el interés, los conflictos y los problemas percibidos sobre el proyecto.

Tabla 2

Matriz de Análisis de Involucrados

Actores Involucrados	Intereses en el Problema Central	Problemas Percibidos	Recursos y Capacidades	Intereses sobre el proyecto	Conflictos Potenciales /Consecuencias del Proyecto
Socios del Barrio	Contar con una mejor información financiera Condiciones de trabajo satisfactorios	Mal registro de información personal y pagos de alícuotas	Recurso Humano	Tener una mejor información financiera y control de pagos.	El sistema no sea usado de la mejor manera
Secretario del Barrio	Analizar el impacto que tienen los socios al momento de recibir información financiera	Realizar los registros manualmente, haciendo de esto una pérdida de tiempo	- Recurso humano - Recurso Tecnológico - Infraestructura	Tener un sistema de control y registro de pagos de alícuotas	No tener conocimientos sobre computación para poder manejarlo adecuadamente al sistema

El interés que tienen los socios sobre el problema es contar con una mejor información financiera ya que disponen de un mal registro de pagos. La principal utilidad del proyecto es tener un informe financiero exacto y el conflicto es que se generaría reportes irreales sin completitud de información.

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL

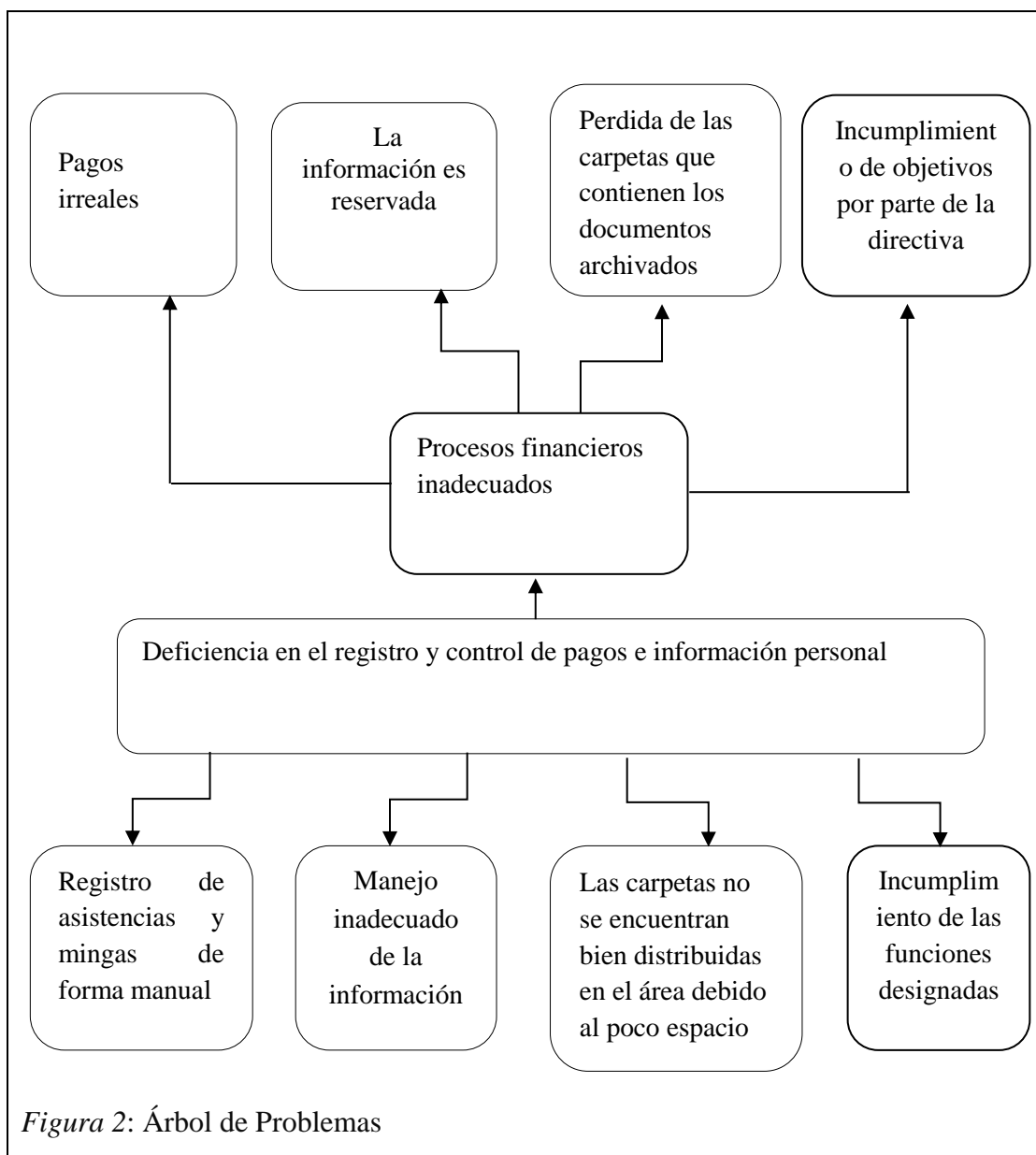


El beneficio que tiene el secretario del barrio sobre el problema es que con la implementación del software la información sea oportuna y exacta y también los socios al momento de recibir un detalle sobre los pagos realizados sea correcto. El beneficio sobre el proyecto es tener un sistema de registro y control de pagos y como conflicto tenemos que la persona que vaya a manejar el software no cuente con los conocimientos necesarios sobre tecnología.

CAPITULO III: PROBLEMAS Y OBJETIVOS

3.01Árbol de Problemas

Es una técnica participativa que ayuda a desarrollar ideas creativas para identificar el problema y organizar la información recolectada, generando un modelo de relaciones causales que lo explican.

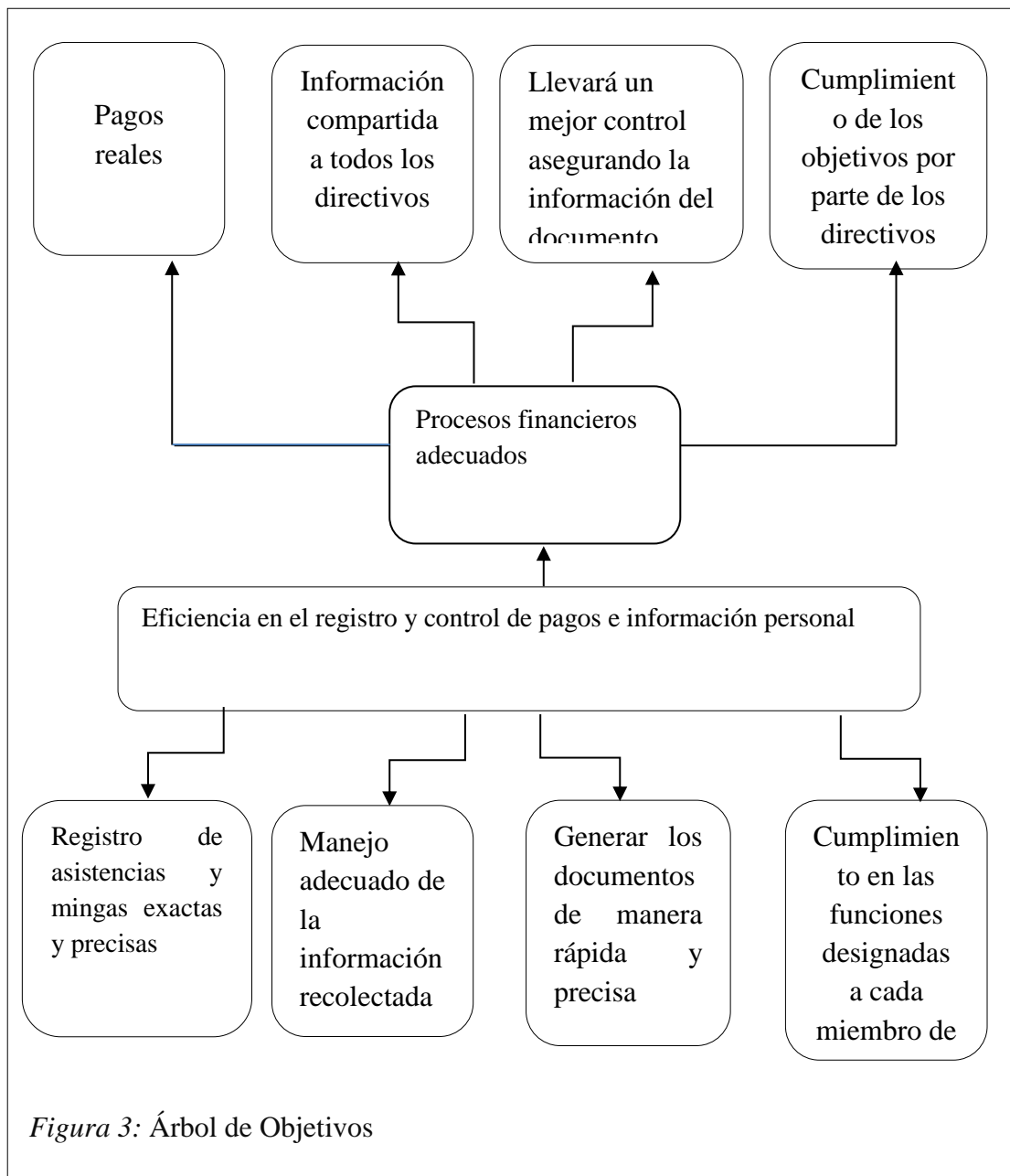


El problema que tiene el barrio es la deficiencia en el registro y control de pagos e información personal y procesos financieros inadecuados para ello tenemos como causas información reservada, el incumplimiento de las funciones designadas a los directivos provocando así falencias en los informes financieros, una administración deficiente y la inobservancia con los objetivos propuestos.



3.02 Árbol de Objetivos

El árbol de objetivos es un procedimiento metodológico que nos permite describir la situación futura que prevalecerá una vez resuelto los problemas e identificar y clasificar los objetivos por orden de importancia y visualizar en un diagrama las relaciones. De este modo, los estados negativos que muestra el “árbol de problemas” se convierten en estados positivos que hipotéticamente se alcanzarán a la conclusión del proyecto





El objetivo principal es la eficiencia en el registro y control de pagos de alícuotas e información personal para así lograr procesos financieros adecuados, teniendo como componentes el manejo adecuado de la información, generar documentos de manera más rápida y precisa con el fin de realizar pagos reales, que la información sea compartida con todos los directivos, llevar un mejor control asegurando el detalle del documento.



CAPITULO IV: ANÁLISIS DE ALTERNATIVAS

4.01 Matriz De Análisis De Alternativas

La matriz de análisis de alternativas es una herramienta que puede emplearse para comparar alternativas utilizando métodos cualitativos o cuantitativos. Consiste en una matriz que contiene las alternativas en las filas y los criterios en las columnas. En cada casilla se anota la valoración de la alternativa respecto al criterio correspondiente, junto con la información que sustenta dicha calificación.

Tabla 3

Matriz de Análisis de Alternativas



MATRIZ DEL ANÁLISIS DE ALTERNATIVAS							
OBJETIVOS	Impacto sobre el propósito	Factibilidad Técnica	Factibilidad Financiera	Factibilidad Social	Factibilidad Política	TOTAL	CATEGORÍAS
Pagos reales	4	4	4	4	4	20	Alta
Información compartida a todos los directivos	3	3	3	2	2	13	Mediana Baja
Llevará un mejor control asegurando la información del documento	3	3	3	3	3	15	Mediana Alta
Cumplimiento de los objetivos por parte de los directivos	4	3	3	2	3	15	Mediana Alta
Generar los documentos de manera rápida y precisa.	4	4	3	3	3	17	Mediana Alta
TOTAL	18	17	16	14	15	80	

En el caso de valoración cualitativa, en lugar de utilizar números, las alternativas se puntúan empleando una escala ordinal del tipo:

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL



Bajo (B) si la alternativa no cumple el criterio;

Medio (M) si la alternativa cumple medianamente el criterio;

Alto (A) si la alternativa cumple totalmente el criterio.

4.02 Matriz de Análisis de impacto de los Objetivos

Tabla 4

Matriz de Análisis de Impacto



Objetivos	Factibilidad de	Impacto en	Impacto	Relevancia	Sostenibilidad	T
	Lograse	Género	Ambiental	(Alta-Media-	ad	o
	(Alta-Media-	(Alta-Media-	(Alta-Media-	Baja)	(Alta-Media-	t
	Baja)	Baja)	Baja)	(4 - 2 -	Baja)	a
	(4 - 2 - 1)	(4 - 2 - 1)	(4 - 2 - 1)	1)	(4 - 2 - 1)	l
Objetivos	4.Los beneficios son mayores a los costos	3. Incrementa las plazas de empleo a las mujeres.	4. Apoya al ambiente reduciendo la utilización de papel.	3. Se cumplen las expectativas desde las más bajas a las más altas.	3. Fortalece la participación de los beneficiarios .	
	4. Cuenta con financiamiento propio	4. Aumenta los ingresos de la mujer.	4. Mejora el entorno social, utilizando tecnología de punta.	4. Es una prioridad concebida por los socios del barrio.	4.A través de autofinanciamiento de los miembros del barrio	8
	4. Es aceptable y conveniente para los beneficiarios del barrio Santa Isabel	3. Mejora el nivel educativo de la mujer.	4. Mejora el entorno cultural, en base al conocimiento informático.	4. Beneficia a los socios, empleados, personal contratado.	4. Los socios están en posibilidades de aportar medios, materiales y económicos	3
	4. Se conoce de la tecnología actualizada para su correcta aplicación y funcionamiento	4. Aumenta la práctica y realización de la mujer	4. Protege la utilización del recurso humano y material	3.Los beneficios son esperados por los directivos del barrio	4. Los directivos están en posibilidades de aportar conocimientos actuales.	o
	16 puntos	15 puntos	20 puntos	16 puntos	16 puntos	



El barrio contratará personal femenino para el manejo del sistema apoyando a mejorar el conocimiento tecnológico, aumentando los ingresos económicos y fortaleciendo los valores éticos y morales de la mujer.

Con la utilización de herramientas informáticas se podrá automatizar el control para los procesos contribuyendo a proteger el entorno ambiental, minimizando la utilización del papel con el fin de mejorar la calidad de vida, defender y restaurar el medio ambiente. Favoreciendo la educación ambiental ya que la naturaleza es una herramienta imprescindible para la supervivencia de la humanidad.

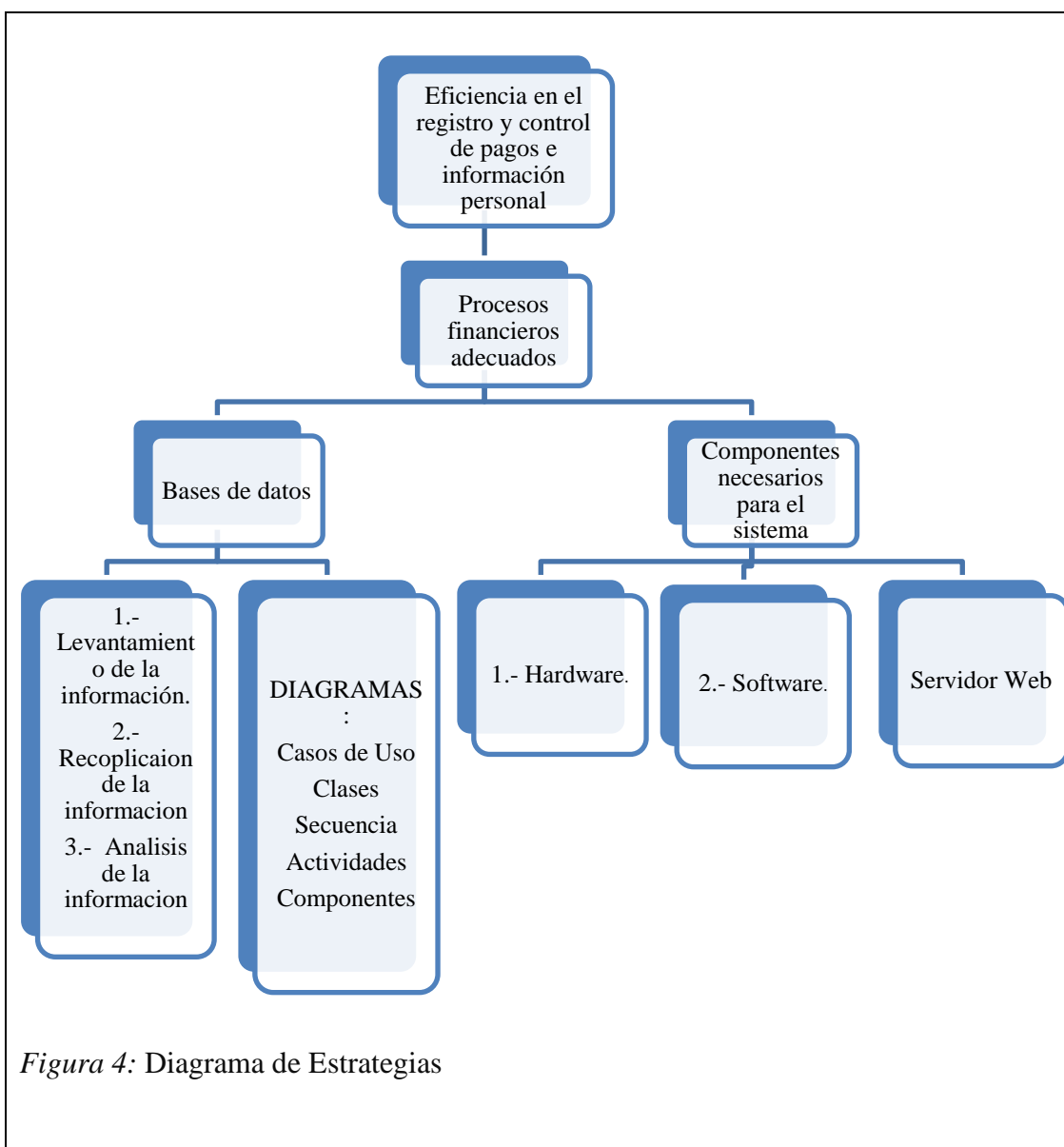
La implementación de un software de registro y control de pagos de alícuotas es de gran importancia ya que controlará de manera eficiente los procesos y procedimientos, cumpliendo con las perspectivas de los beneficiarios, favoreciendo a socios, empleados, personal de trabajo.

Para que el sistema funcione correctamente es necesario que el personal esté capacitado para su manejo, esto ayudará a que el barrio este organizado internamente y sus procesos se realicen con seguridad y rapidez. El sistema se mantendrá activo en el transcurso del tiempo que la organización desee teniendo un financiamiento a futuro.

4.03 Diagrama de Estrategias

El diagrama tiene como finalidad definir la estructura y alcance de estrategias, vinculando con los objetivos que puedan ser trabajados de forma articulada determinando así los límites y objetivos específicos del proyecto.

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



4.04 Matriz de Marco Lógico

La matriz del marco lógico nos permite saber en síntesis que es lo que se desea lograr en el proyecto, como se alcanzara el propósito trazado y sus componentes, como se mide el éxito de los objetivos, resultados y que recursos se necesitan para la ejecución del proyecto.

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



El Fin es una definición de cómo el proyecto o programa contribuirá a la solución del problema (o problemas) del sector.

Los indicadores a nivel de Fin miden el impacto general que tendrá el proyecto. Son específicas en términos de cantidad, calidad y tiempo. (grupo social y lugar, si es relevante).

Los medios de verificación son las fuentes de información que se pueden utilizar para verificar que los objetivos se lograron. Pueden incluir material publicado, inspección visual, encuestas por muestreo, etc.

Los supuestos indican los acontecimientos, las condiciones o las decisiones importantes necesarias para la "sustentabilidad".

El Propósito es el impacto directo a ser logrado como resultado de la utilización de los Componentes producidos por el proyecto.

Los indicadores a nivel de Propósito describen el impacto logrado al final del proyecto. Deben incluir metas que reflejen la situación al finalizar el proyecto.

Los medios de verificación son las fuentes que el ejecutor y el evaluador pueden consultar para ver si los objetivos se están logrando.

Los supuestos indican los acontecimientos, las condiciones o las decisiones que tienen que ocurrir para que el proyecto contribuya significativamente al logro del Fin.



Los Componentes son las obras, servicios, y capacitación que se requiere que complete el ejecutor del proyecto de acuerdo con el contrato. Estos deben expresarse en trabajo terminado (sistemas instalados, gente capacitada, etc.)

Los indicadores de los Componentes son descripciones breves, pero claras de cada uno de los Componentes que tiene que terminarse durante la ejecución.

Los medios de verificación indican dónde el evaluador puede encontrar las fuentes de información para verificar que los resultados que han sido contratados han sido producidos.

Los supuestos son los acontecimientos, las condiciones o las decisiones que tienen que ocurrir para que los componentes del proyecto alcancen el Propósito para el cual se llevaron a cabo.

Las Actividades son las tareas que el ejecutor debe cumplir para completar cada uno de los Componentes del proyecto y que implican costos.

Los indicadores de las actividades contiene el presupuesto para cada Componente a ser producido por el proyecto.

Los medios de verificación indican donde un evaluador puede obtener información para verificar si el presupuesto se gastó como estaba planeado



Los supuestos son los acontecimientos, condiciones o decisiones (fuera del control del gerente de proyecto) que tienen que suceder para completar los Componentes del proyecto. Matriz de Marco Lógico (**VER ANEXO A01**)



CAPÍTULO V: Propuesta

Justificación Técnica

El presente documento está elaborado con el fin de controlar los procesos y procedimientos, que se ejecutan dentro del sistema de registro de pagos de alcúotas a fin de optimizar los recursos, tiempos de respuesta y actividades que el barrio Santa Isabel realiza.

Las aplicaciones a utilizarse en la elaboración del software son:

Servidor: SQL Server 2008 R2, modelamiento UML: Rational Rose, desarrollo: Visual Studio.

Aplicar una metodología orientada a objetos que permita conjugar todas las tareas que encierra el desarrollo de una aplicación web, ha constituido una investigación extensa y compleja, pero a decir de la estructuración misma lo desarrolla RUP (Proceso Unificado Racional), esta metodología en sus fases correspondientes establecen claramente lo que ha significado la construcción de la presente aplicación; su diseño está enfocado a un ambiente netamente gráfico, para el hemos utilizado UML(Lenguaje de Modelamiento Unificado), lo que ha permitido poder entender los procesos, procedimientos y actividades que se cumplen en cada una de las áreas del barrio, y a su vez las relaciones de información que mantienen cada una de ellas.

Las actividades que se encuentran materializadas en la esquematización del diseño global determinan la utilización de diagramas como:

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



- Diagrama de Caso de Uso
- Diagramas de Realización
- Diagrama de Secuencias
- Diagrama de Colaboración
- Diagrama de Componentes
- Diagrama de Clases
- Modelo Lógico
- Modelo físico

Este documento encaminará al Técnico a visualizar los requerimientos respectivos de las especificaciones de diseño del sistema para determinar la organización y distribución de los diferentes modelos lógicos y físicos que dispone la plataforma virtual.

La Arquitectura utilizada en el sistema es de tres capas, (presentación, datos y negocio) en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño. La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

La implementación de arquitectura y estándares de diseño ya mencionadas nos ayudarán a construir y documentar un sistema de forma adecuada, para cumplir con los requisitos establecidos, la programación orientada a objetos y la aplicación de sus características fusionan el complemento ideal de una aplicación orientada a la web.



5.02 Análisis y Diseño

5.02.01 Diagrama de Caso de Uso General

El Diagrama de Casos de Uso muestra un conjunto de actividades, actores implicados y sus relaciones. Este diagrama es fundamental ya que nos ayuda a visualizar el funcionamiento en el modelado y organización del sistema.

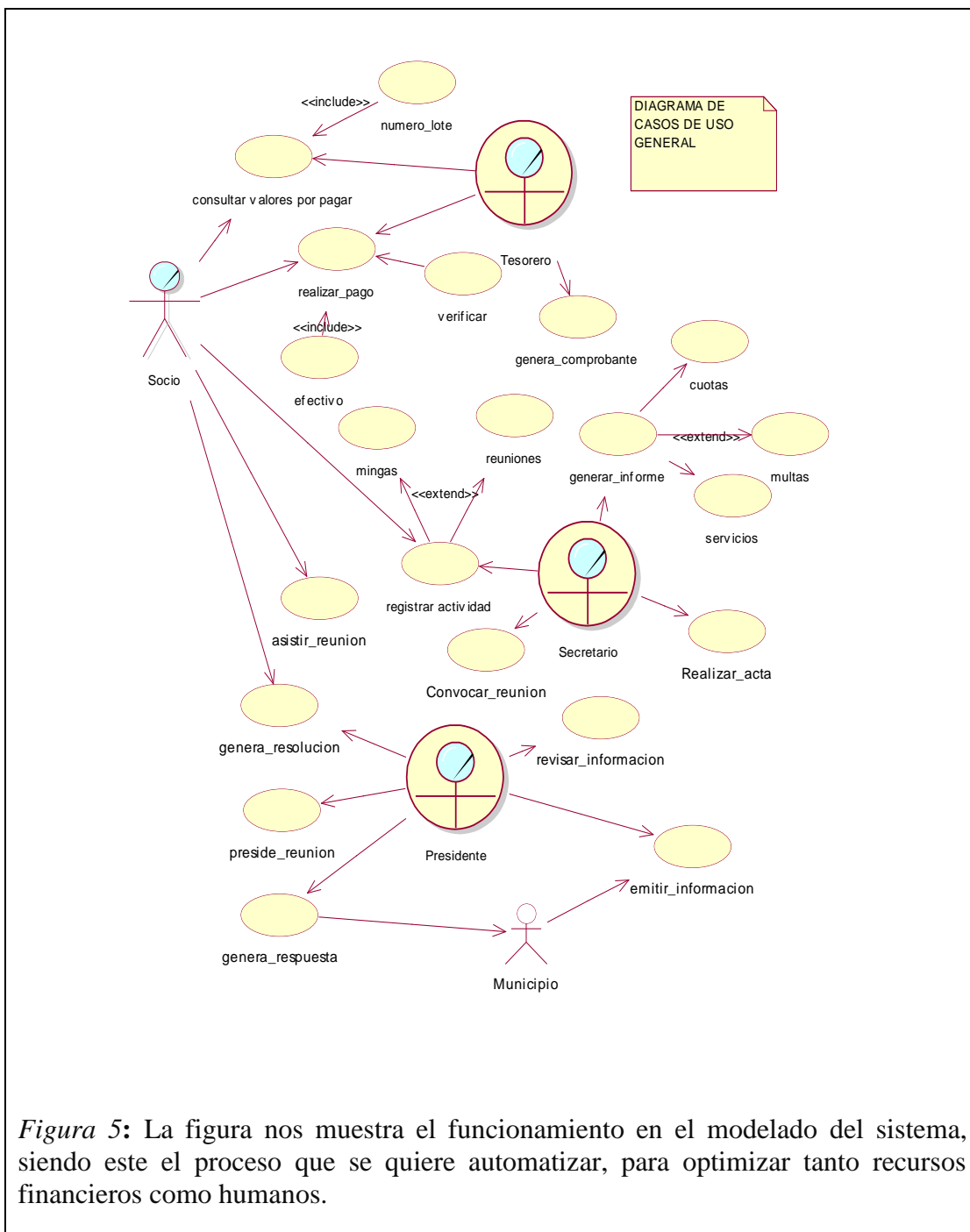


Figura 5: La figura nos muestra el funcionamiento en el modelado del sistema, siendo este el proceso que se quiere automatizar, para optimizar tanto recursos financieros como humanos.

5.02.02 Diagramas de Realización

Un diagrama de realización representa los flujos de trabajo paso a paso de negocio y operacionales de los componentes en un sistema.

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL

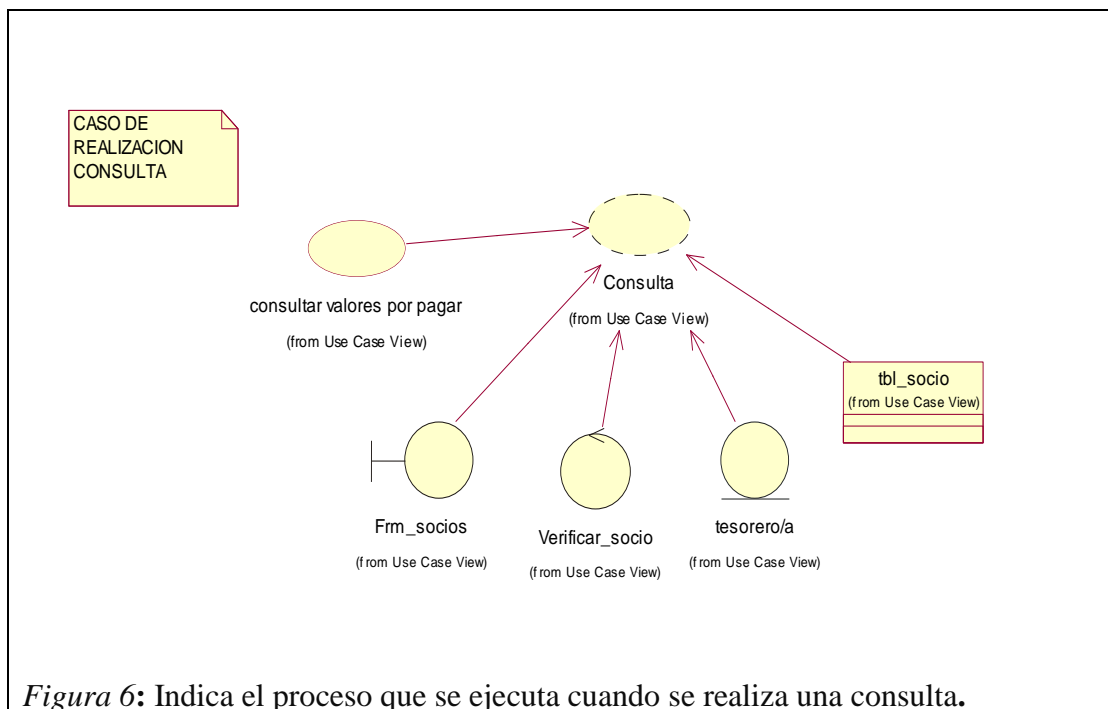


Tabla 5

Caso de Uso de la consulta de valores a pagar

Especificación del caso de uso: el socio consulta valores por pagar		
Código	1	
Nombre	Consultar valores por pagar	
Descripción	Este caso de uso permite que el socio conozca el valor adeudado.	
Autores	Alicia	
Fecha creación	Marzo-2014	Fecha última modificación Marzo 26-2014
Actores	Socio, tesorero	
Precondición	El socio tiene que proporcionar el número de lote	
Pos condición	El socio entrega la información requerida.	
Flujo normal	1. Ingresar nombre de usuario 2. Ingresar contraseña 3. Validar datos 4. Realizar consulta	
Excepciones	Ninguna	
Anotaciones	Ninguna	

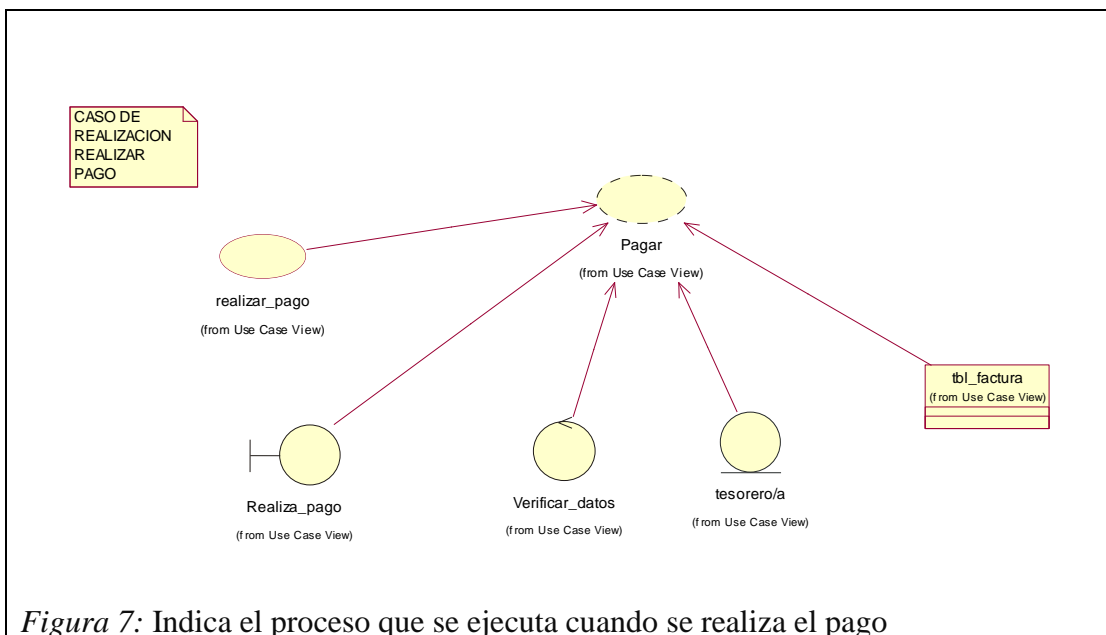


Figura 7: Indica el proceso que se ejecuta cuando se realiza el pago

Tabla 6

Caso de uso de realizar pago

Especificación del caso de uso: realizar pago		
Código	2	
Nombre	Realizar Pago	
Descripción	Este caso de uso permite registrar el pago que realice el socio	
Autores	Alicia	
Fecha creación	Marzo- 2014	Fecha última modificación Marzo 26-2014
Actores	Socio, tesorero	
Precondición	El tesorero realiza la consulta correspondiente	
Pos condición	Registra el pago.	
Flujo normal	1. Solicitar datos del socio 2. Verificar datos 3. Registrar pago	
Excepciones	Ninguna	
Anotaciones	Ninguna	

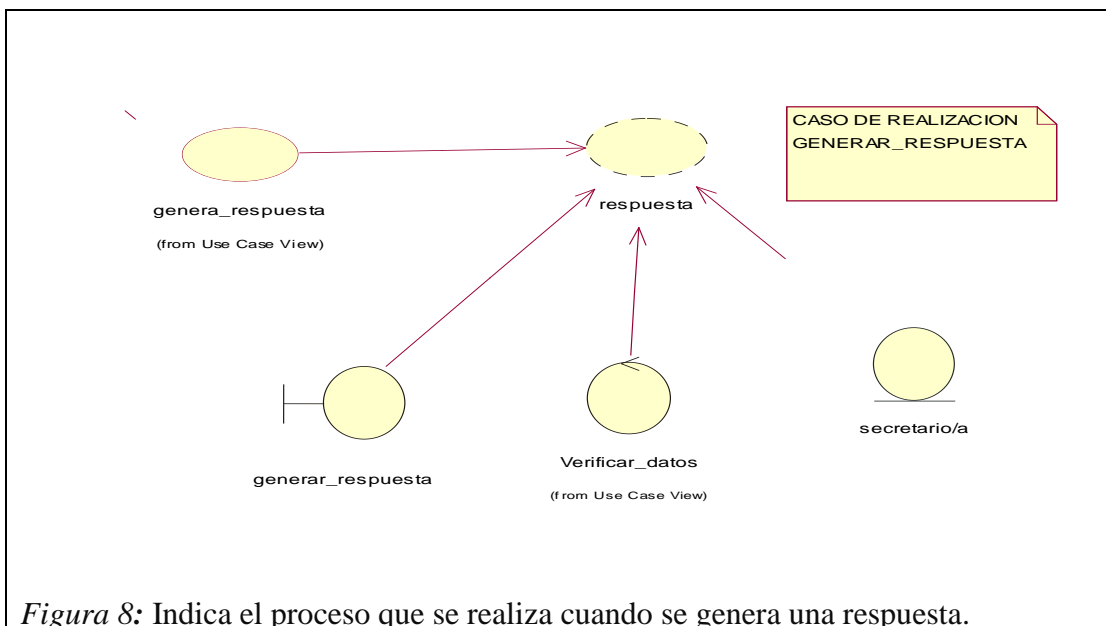


Figura 8: Indica el proceso que se realiza cuando se genera una respuesta.

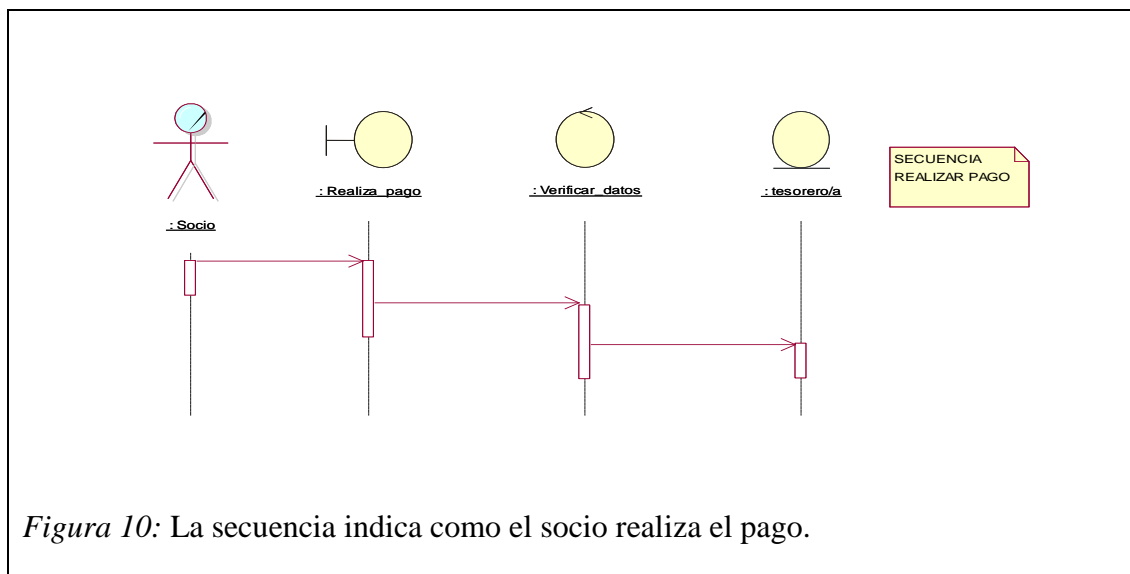
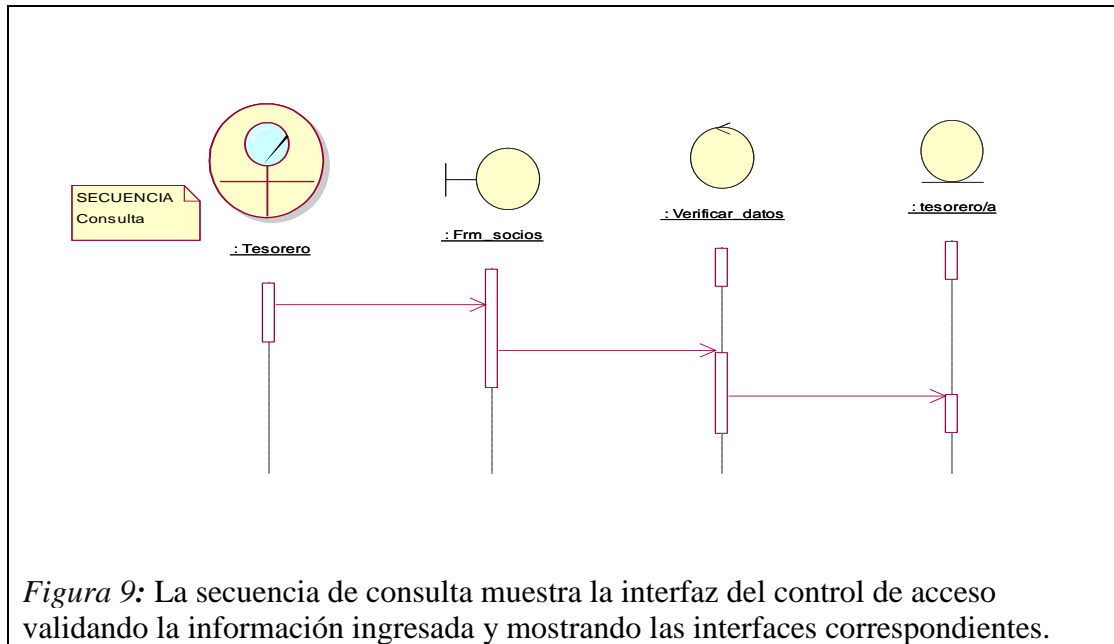
Tabla 7

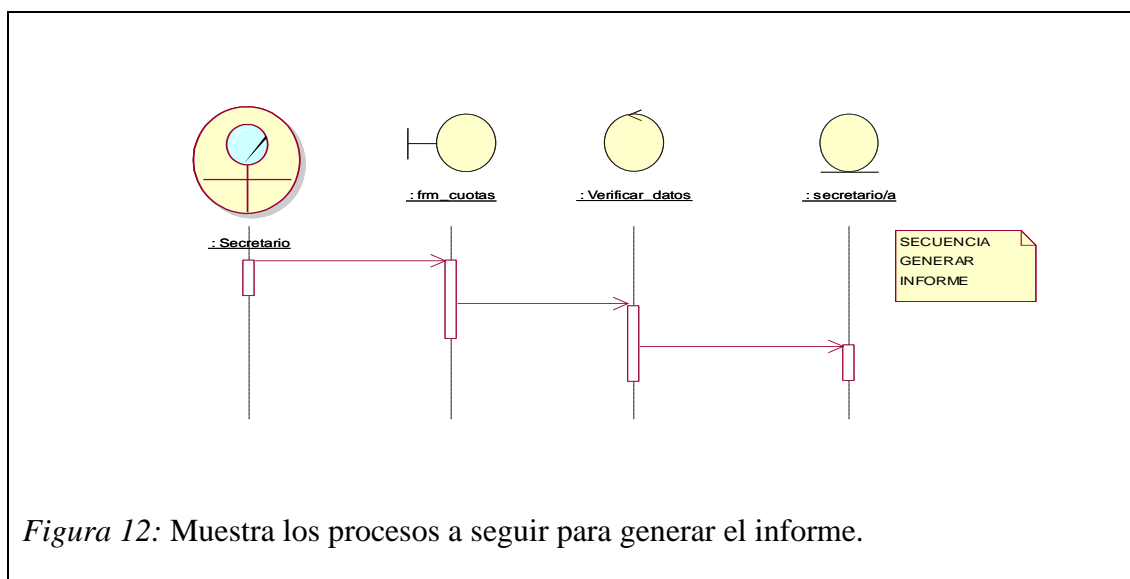
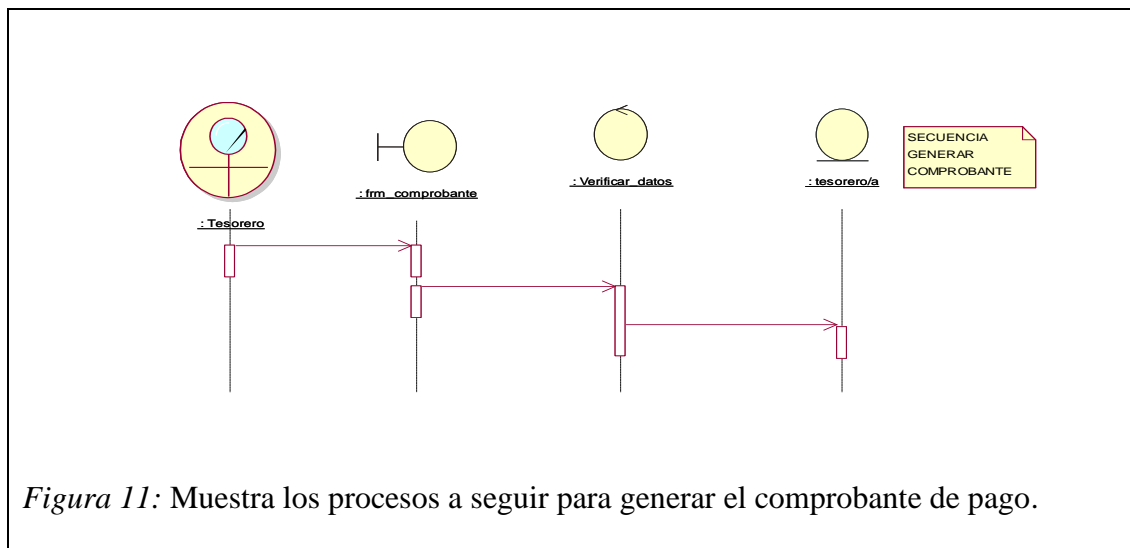
Caso de uso de generar respuesta

Especificación del caso de uso: el presidente genera una respuesta por un oficio recibido		
Código	7	
Nombre	Generar_respuesta	
Descripción	Este caso de uso permite al presidente generar una respuesta por un oficio recibido	
Autores	Alicia	
Fecha creación	Marzo-2014	Fecha última modificación Marzo 26-2014
Actores	Socio, tesorero	
Precondición	El presidente recibe oficio	
Pos condición	El presidente emite oficio	
Flujo normal	1. El presidente recibe un oficio 2. Revisa la información 3. Emite la respuesta	
Excepciones	Ninguna	
Anotaciones	Ninguna	

5.02.03 Diagramas de Secuencia

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso.





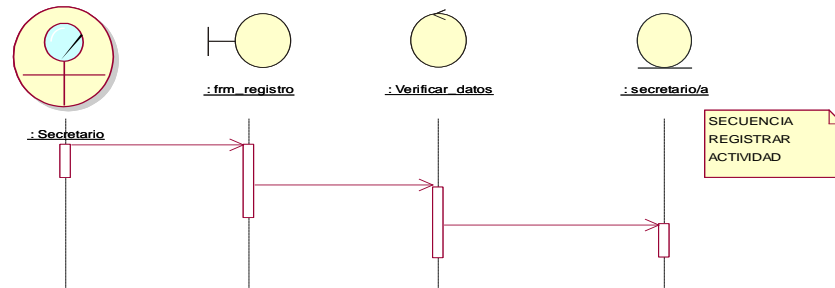


Figura 13: La secuencia indica el registro de la actividad realizada.

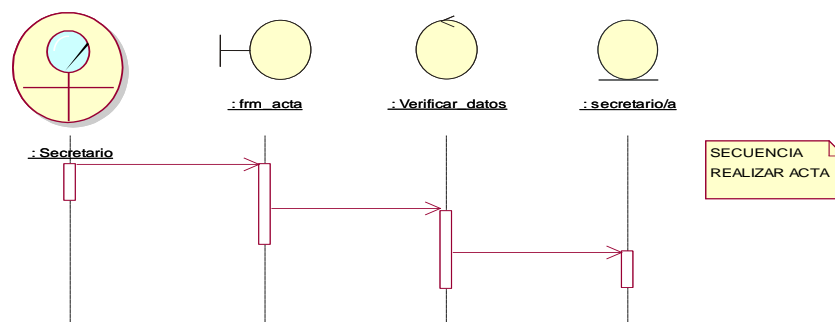


Figura 14: Muestra los procesos a seguir para realizar el acta generada en la asamblea.

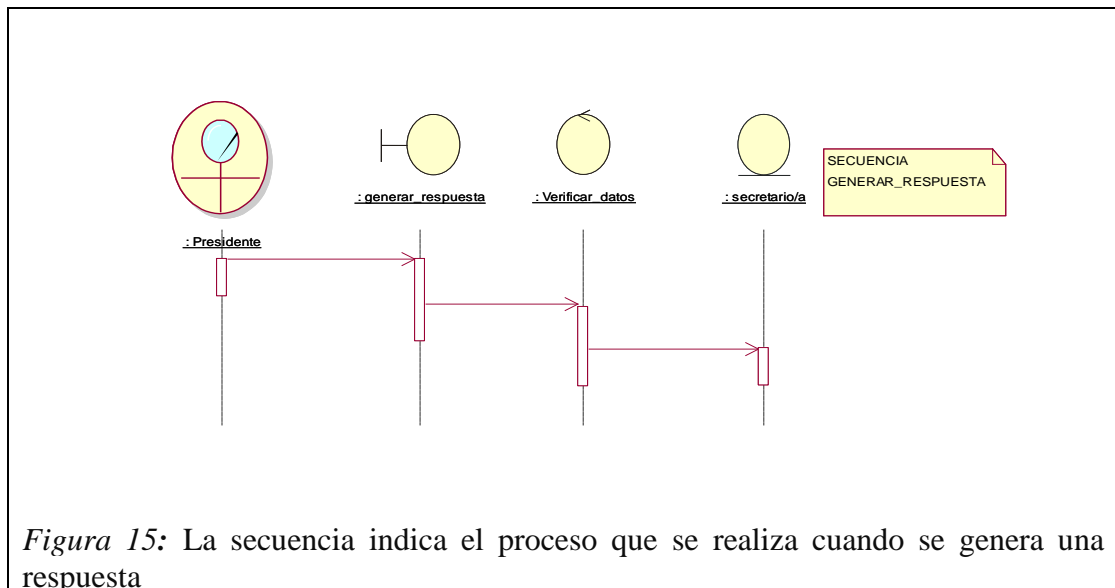


Figura 15: La secuencia indica el proceso que se realiza cuando se genera una respuesta

5.02.04 Diagrama de Colaboración

Un diagrama de colaboración es esencialmente un diagrama que muestra interacciones organizadas alrededor de los objetos. A diferencia de los diagramas de secuencia, los diagramas de colaboración, también llamados diagramas de comunicación, muestran explícitamente las relaciones de los roles.

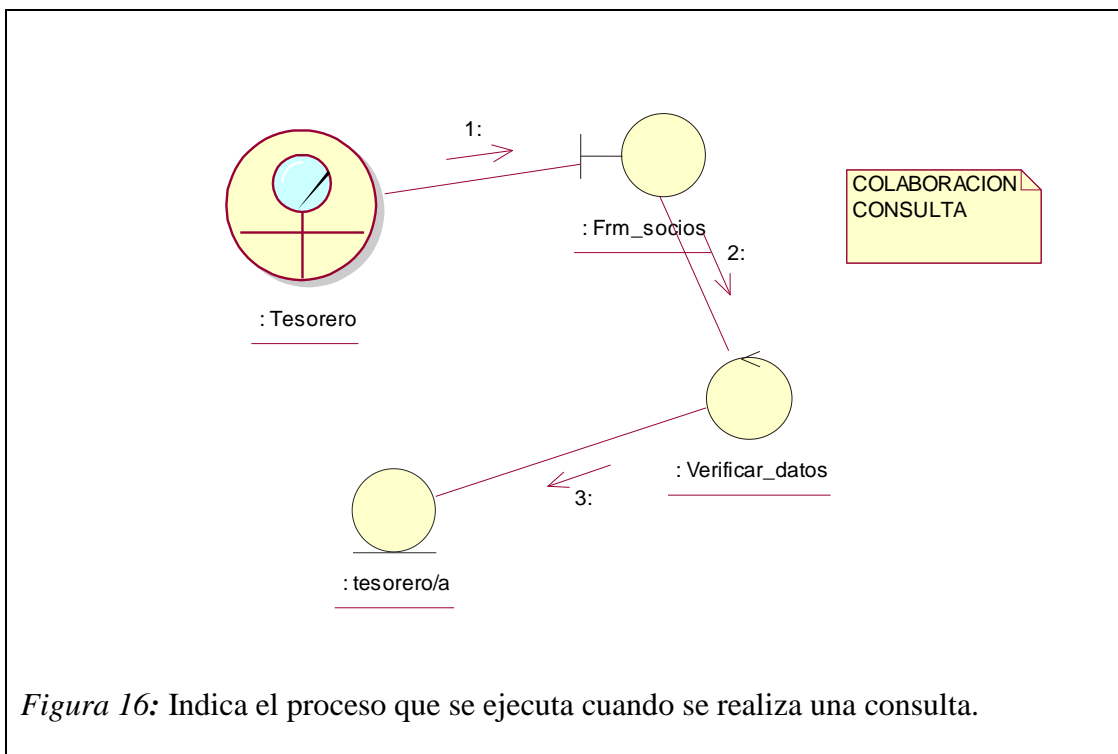


Figura 16: Indica el proceso que se ejecuta cuando se realiza una consulta.

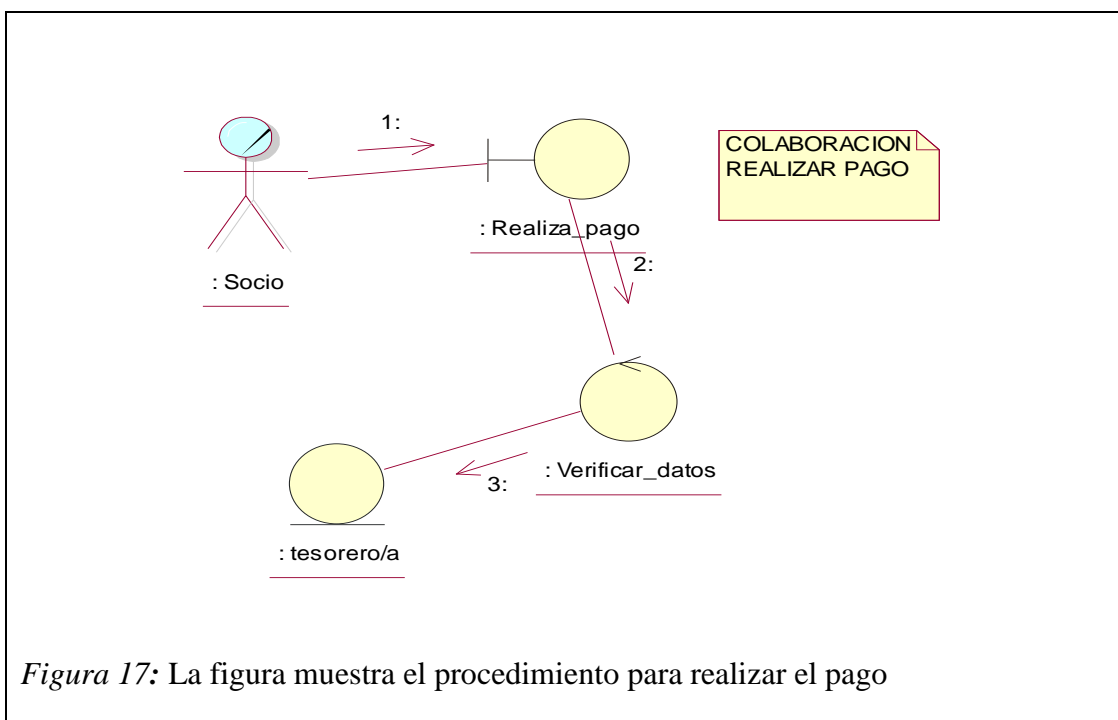


Figura 17: La figura muestra el procedimiento para realizar el pago

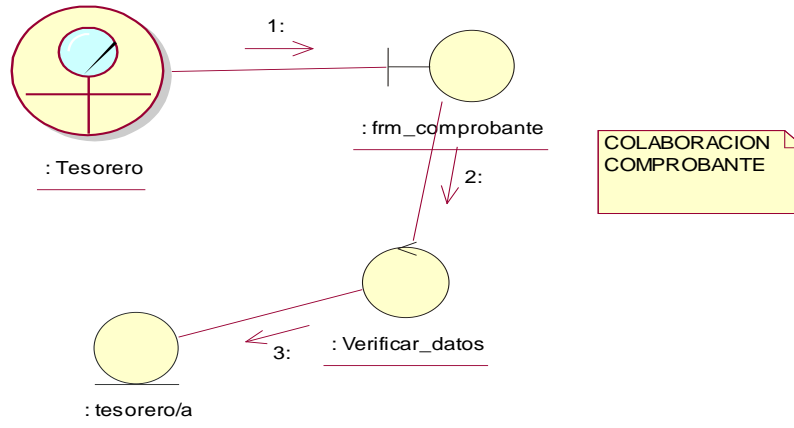


Figura 18: Indica el proceso que se realiza para generar el comprobante de pago

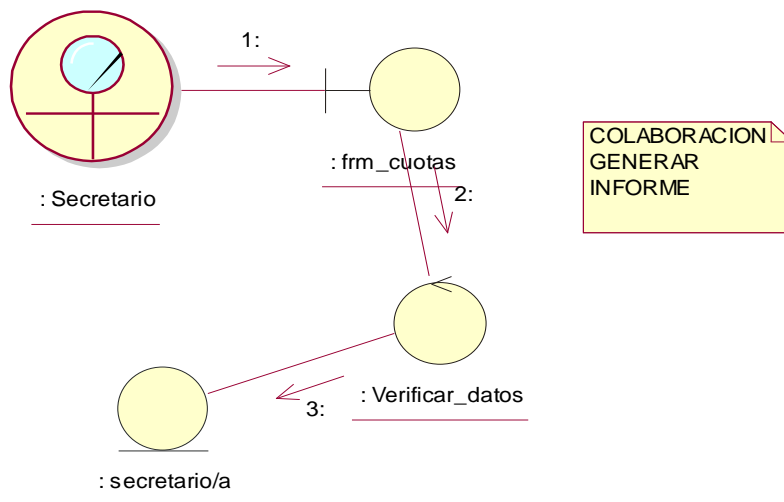
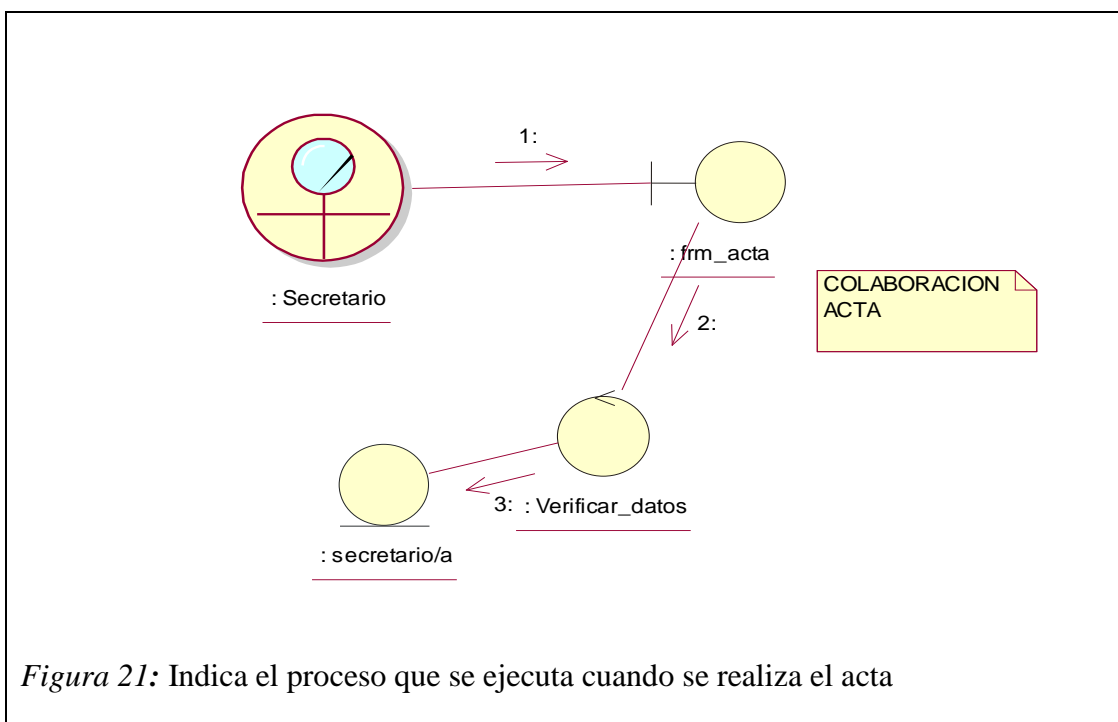
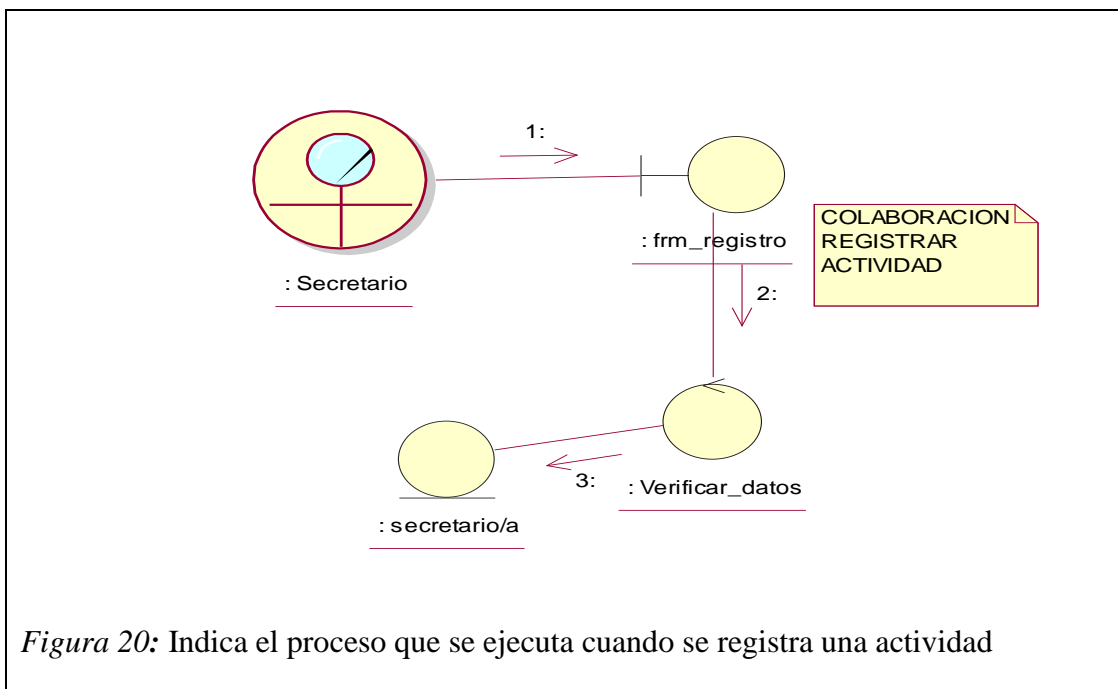
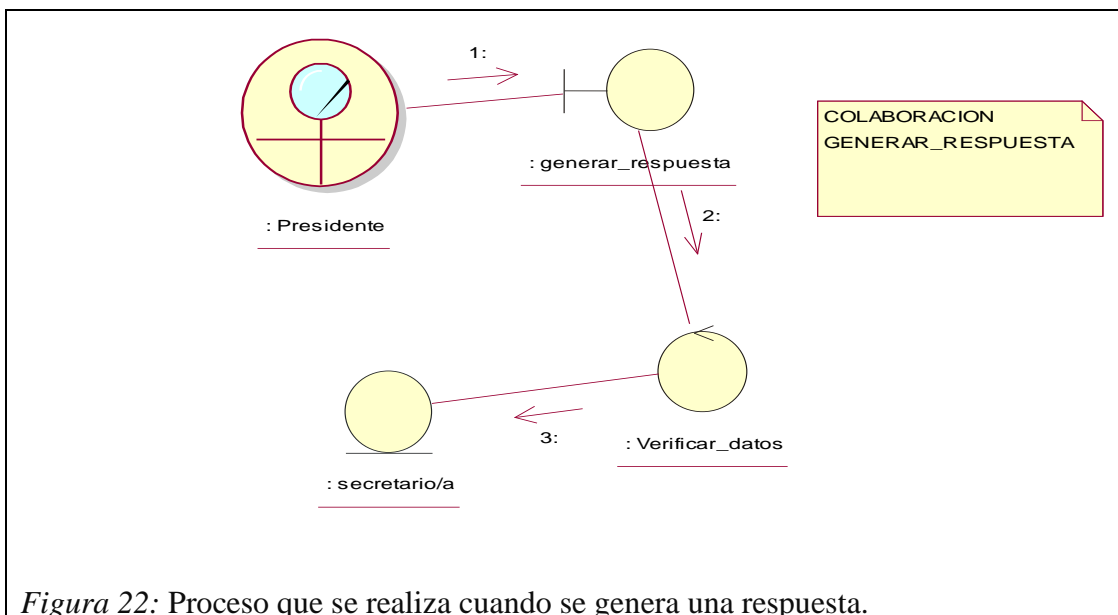


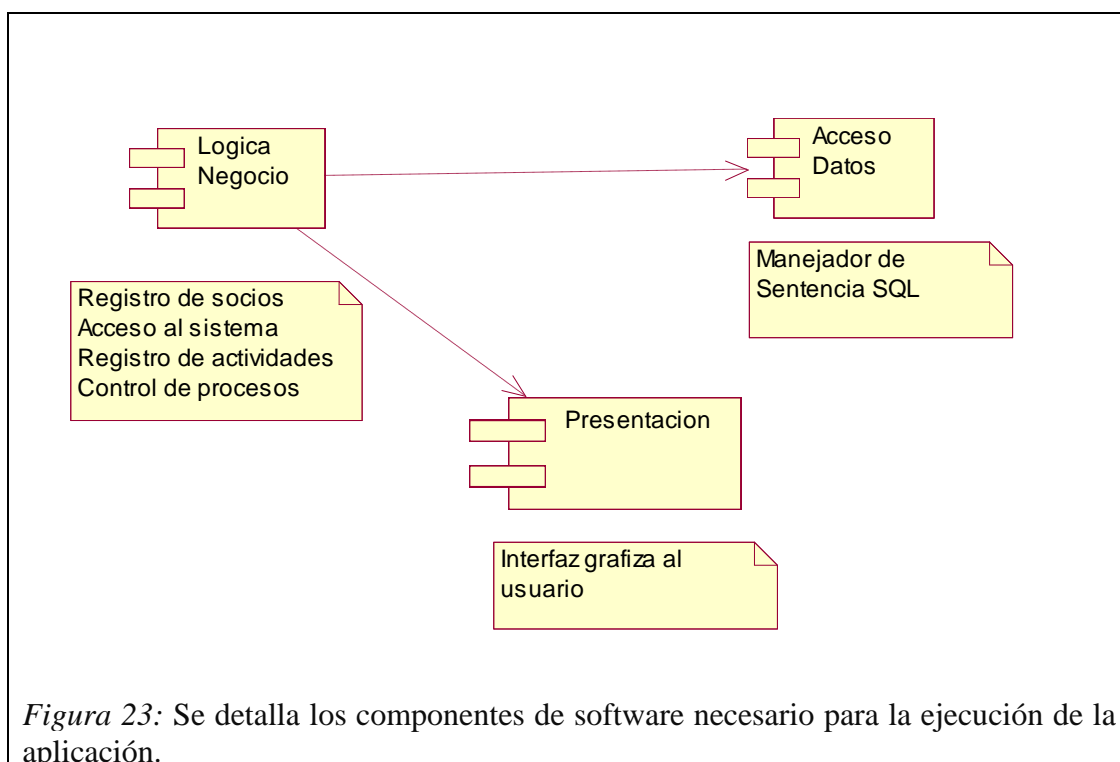
Figura 19: Proceso que se realiza para generar un informe de pagos





5.02.05 Diagrama de Implementación

Los diagramas de implementación ofrecen una ilustración de la arquitectura física del hardware, del software y de los artefactos del sistema. Los diagramas de implementación pueden entenderse como lo contrario de los casos de uso, porque ilustran la forma física del sistema, en lugar de representar conceptualmente los usuarios y dispositivos que interactúan con el sistema





5.02.06 Diagrama de Clases

Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de agregación, ya que una clase es una descripción de conjunto de objetos que comparten los mismos atributos, operaciones, métodos, relaciones y semántica; mostrando un conjunto de elementos que son estáticos, como las clases y tipos junto con sus contenidos y relaciones. Un diagrama de clases está compuesto por los siguientes elementos: Clase: atributos, métodos y visibilidad. Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

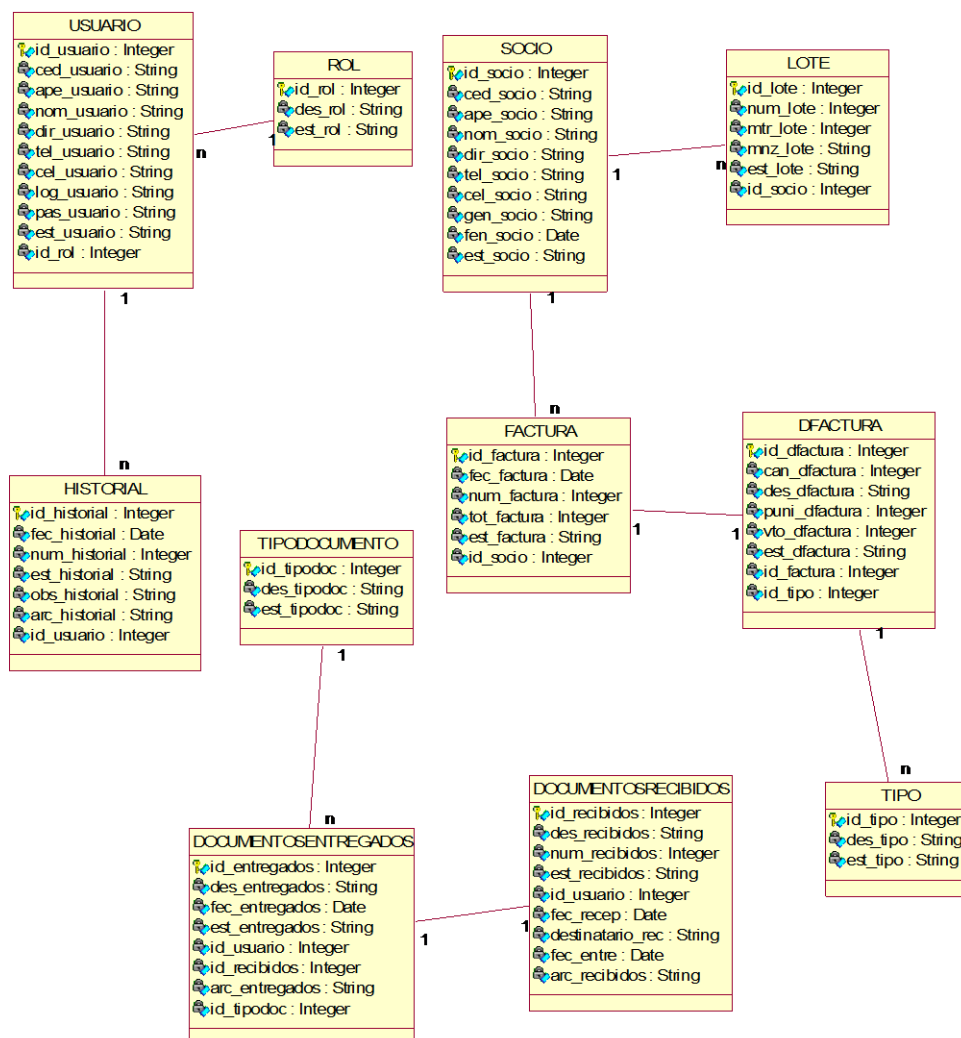


Figura 24: La figura muestra un diagrama que recopila toda la información necesaria para el diagrama físico de flujo de datos, este esquema es la base principal para el modelamiento y creación de la base de datos.

5.02.07 Modelo Lógico

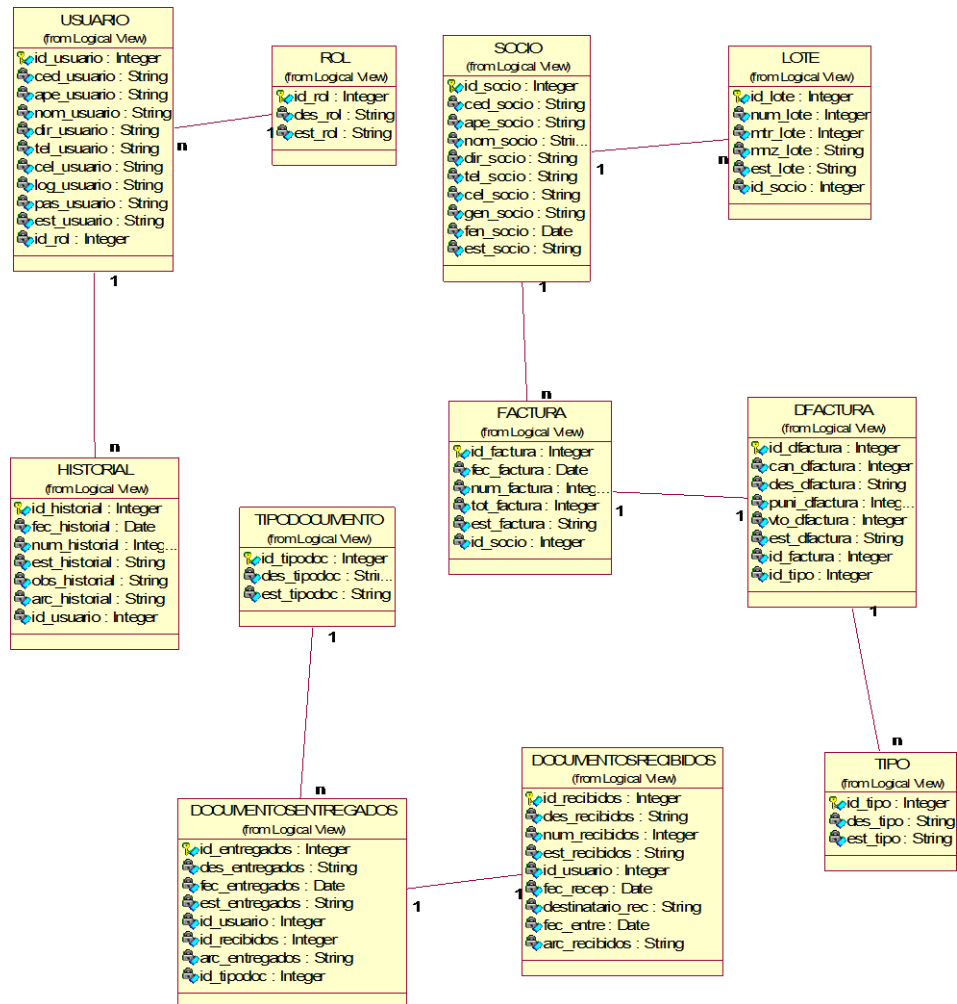


Figura 25: La figura muestra el modelo lógico del sistema, en donde se encuentran las tablas correspondientes a la base de datos la cual almacena los datos para la utilización de la aplicación.

5.02.08 Modelo Físico

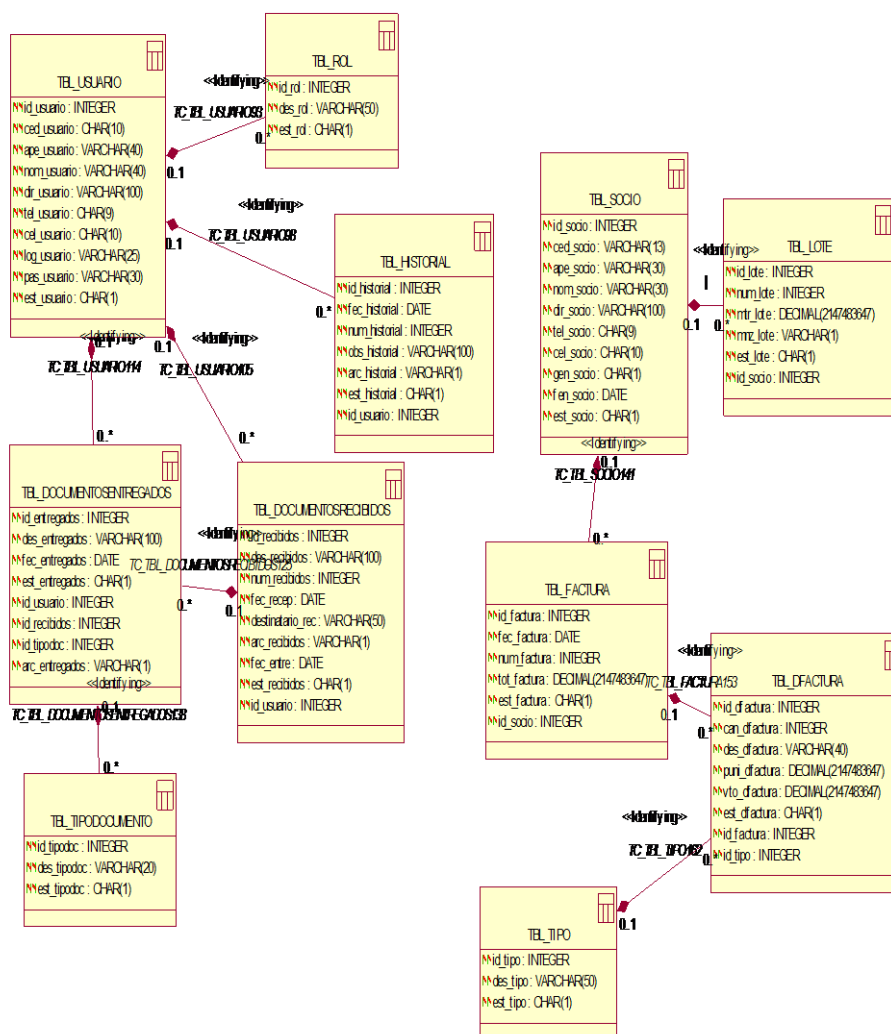


Figura 26: La figura representa el modelamiento de la base de datos.



5.03 Desarrollo

5.03.01 Arquitectura Del Sistema

El proyecto está diseñado y desarrollado para que su funcionamiento sea en tres capas o conocida como orientada a la web, las capas utilizadas son tres.

5.03.01.01 Capa de Presentación:

Es la capa de presentación para el usuario, donde brinda y recopila información del usuario realizando varios pasos.

Es también conocida como interfaz gráfica donde es caracterizada por ser amigable, fácil de utilizar y entendible para el usuario. Esta capa tiene únicamente comunicación con la capa de negocio.

5.03.01.02 Capa de Negocio

Es donde se encuentran los programas que se ejecutan, se recibe la información y envía las respuestas de las peticiones del usuario en el proceso. Se le denomina capa o lógica del negocio, aquí se establecen todas las reglas a cumplir. Esta capa tiene conexión con la capa de presentación donde desde ahí se reciben las peticiones y se brinda resultados al usuario, también está conectada con la capa de base de datos la cual permite almacenar, modificar, eliminar y recuperar datos.

5.03.01.03 Capa de Datos:



Es donde se encuentra toda la información almacenada, la que está formada por uno o más sistemas administradores que realizan lo referente al almacenamiento de datos o recuperación de información desde la capa de negocio.

5.03.02 Módulos

5.03.02.01 Módulo de Seguridad

El módulo de seguridad permite identificar al usuario que va a manipular el sistemas, se asignara los roles correspondientes a cada usuario para generar un mejor nivel de seguridad en cada perfil.

El usuario podrá tener los privilegios del sistema y manipularlo siempre y cuando su rol sea de administrador, existen solo dos roles en el sistema como son: Administrador y Técnico.

5.03.02.02 Módulo de Mantenimiento

El módulo de Mantenimiento es el encargado Buscar, Agregar, Modificar y Eliminar la información registrada en la Base de Datos

Los mantenimientos son a:

- Usuarios
- Rol de usuario
- Socios
- Documentos Entregados



- Documentos recibidos
- Factura

Estos mantenimientos de encuentran categorizados de acuerdo al nivel de seguridad que se le asigne a los usuario. Para su correcta manipulación.

6.05.06Módulo Lógica Negocios

Permite crear las validaciones de los procesos que realiza el sistema con la finalidad de que exista integridad en la información.

5.03.03 Estándares

5.03.03.01 Estándares de Programación

Un estándar de programación es una forma de regularizar la programación de tal forma que al momento de trabajar con un proyecto sea entendido que pueda acceder sin ningún inconveniente al revisar el código.

5.03.03.01.01 Declaración de Variables

Nombre que se asigna a la variable para poder identificarla y deberá estar asociada al motivo para declararla.

Ejemplo: `TBL_USUARIO` UsuarioInfo = `new TBL_USUARIO()`;



5.03.03.01.02 Variables de Tipo Arreglo

En la declaración de arreglos de elementos se declara la variable con el prefijo de “lista”, el cual nos da a entender que es una variable de tipo arreglo la cual contendrá de cero a el número de datos declarados.

Ejemplos: `List<TBL_USUARIO> UsuarioLista = new List<TBL_USUARIO>();`

5.03.03.01.04 Declaración de variables, atributos y objetos

Tabla 8

Declaración de variables

Sintaxis	Ejemplo	Descripción
[TipoVariable] [Nombre de la Variable]	<code>public string nombre;</code> <code>public string apellido;</code> <code>public int contador;</code>	Indica una variable o atributo que guardará un nombre, apellido , contador

5.03.03.01.04 Declaración de Controles

Tabla 9

Tipo de Controles



Tipo de control	Prefijo	Ejemplo
Label	Lbl	lbl_nombre
textBox	Txt	txt_nombre
Button	Btn	btn_aceptar
radioButton	Rdo	rdo_seleccion
checkBox	Chk	chk_seleccion
Imagen	Img	img_nuevo
linkButton	Lnk	lnk_regresar
calendarExtender	Cal	cal_fecha
dataGridView	Dgv	dgv_usuario
hiperLink	Hlink	hlnk_menu
dropDownList	Ddl	ddl_usuarios

5.03.03.01.06 Declaración de Datos

Tabla 10

Tipo de Datos

Tipo de variable	Declaracion	Descripción
Integer	Int	Entero de 32 bits con signo.
Char	Char	Un carácter UNICODE de 16 bits
String	String	Cadena de caracteres
Date	Date	Formato de fecha/hora
Boolean	Bool	Valor lógico: verdadero y falso
Float	Float	Coma flotantes, 11-12 dígitos significativos.
Double	Double	Coma flotante, 64 bits (15-16 dígitos significativos)
Object	Object	Objeto genérico

5.03.03.01.07 Declaración de Clases

Tabla 11

Declaración de clases

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



Sintaxis	Ejemplo	Descripción
[Tipo] class [Nombre de Clase]	<code>public class Logica_Usuarios</code>	Indica una clase usuarios

5.03.03.01.08 Declaración de Funciones y Procedimientos

El nombre de la funciones debe de ser descriptivo a la acción que realiza, debe contener tipo de acceso, dato de retorno, nombre de la función, y argumentos si los necesitase.

Tabla 12

Nombres de Funciones y Procedimientos

Descripción	Nomenclatura
Obtener lista de usuarios	<code>public static List<TBL_USUARIO> ObtenerUsuariosLista()</code>
Obtener la información de usuario por código.	<code>public static TBL_USUARIO ObtenerUsuarioXId(int id)</code>

5.03.03.01.08 Declaración de Métodos

Tabla 13

Declaracion de Métodos



Sintaxis	Ejemplo	Descripción
[Tipo] void nombreProcedimiento[(ListaParámetros)]	<code>private void CargarDatosFormulario(int id)</code>	Indica un método cargar datos de un formulario usuario que recibe una variable por valor de tipo int al ámbito de la clase

5.03.03.02 Estándares de Base de Datos

Los estándares de base de datos deben estar documentados con las siguientes propiedades:

Descripción: debe contener la descripción del uso o la función del objeto y que no sea una repetición del nombre.

5.03.03.02.02 Tipo de Datos

La descripción de los tipos de datos debe ser clara y precisa para su utilización en la base de datos.

Tabla 14

Tipo de Datos



Tipo de Datos	Característica
Bit ó Bool	Un número entero que puede ser 0 ó 1
Integer, Int	Número entero con o sin signo. Con signo el rango de valores va desde -2147483648 a 2147483647. Sin signo el rango va desde 0 a 429.4967.295
Float	Número pequeño en coma flotante de precisión simple. Los valores válidos van desde -3.402823466E+38 a -1.175494351E-38, 0 y desde 1.175494351E-38 a 3.402823466E+38.
Double	Número en coma flotante de precisión doble. Los valores permitidos van desde -1.7976931348623157E+308 a -2.2250738585072014E-308.
Date	Tipo fecha, almacena una fecha. El rango de valores va desde el 1 de enero del 1001 al 31 de diciembre de 9999. El formato de almacenamiento es de año-mes-día
DateTime	Combinación de fecha y hora. El rango de valores va desde el 1 de enero del 1001 a las 0 horas, 0 minutos y 0 segundos al 31 de diciembre del 9999 a las 23 horas, 59 minutos y 59 segundos. El formato de almacenamiento es de año-mes-día horas: minutos: segundos
Time	Almacena una hora. El rango de horas va desde -838 horas, 59 minutos y 59 segundos a 838, 59 minutos y 59 segundos. El formato de almacenamiento es de 'HH:MM:SS'
Char(n)	Almacena una cadena de longitud fija. La cadena podrá contener desde 0 a 255 caracteres.
VarChar(n)	Almacena una cadena de longitud variable. La cadena podrá contener desde 0 a 255 caracteres.

5.03.03.02.02 Tablas

Para los nombres de las tablas de rompimiento se considerará el nombre de las tablas involucradas en el rompimiento tomando las tres primeras letras de cada tabla o si llegara a coincidir las tres primeras El nombre de la tabla se escribirá en mayúsculas y los atributos en minúsculas ya que el gestor de base de datos SQLSERVER al

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



momento de generar la base de datos los escribe así, y se escribirá en singular los nombres de las tablas.

Tabla 15

Nombres de las tablas de la base de datos

Nombre	Nomenclatura
Tabla de usuario	TBL_USUARIO
Tabla roles	TBL_ROL

5.03.03.03 Estándares de Diseño

5.03.03.03.01 Estándares de Diseño UML

(VER ANEXO 02)

5.05 Diseño de Interfaces

Tabla 16

Ingreso al sistema

Ítem	Representación	Descripción
A	Caja de Texto	Usuario
B	Caja de Texto	Clave
C	List Box	Rol
D	Botón	Ingresar

Figura 27: Ingreso al sistema

Tabla 17

Ingreso de un Nuevo Socio

Ítem	Representación	Descripción
A	Caja de Texto	Cedula
B	Caja de Texto	Apellido
C	Caja de Texto	Nombre
D	DropDownList	Genero
E	Caja de Texto	Direccion
F	Caja de Texto	Telefono
G	Caja de Texto	Celular
H	Caja de Texto	Fecha de nacimiento

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL



Tabla 18

Lista de Socios Ingresados

Ítem	Representación	Descripción
A	Botón	Nuevo
B	Caja de Texto	Buscar
C	Botón	Editar
D	Botón	Eliminar



Figura 29: Listado de los socios registrados.

Tabla 19

Agregar Clientes

Ítem	Representación	Descripción
A	Caja de Texto	Fecha
B	Caja de Texto	Observacion
C	D	Usuario

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



Figura 30: Ingreso de nueva acta.

5.05 Pruebas de módulo.

5.05.01Módulo de Seguridad

Este tipo de prueba se realizara mediante la validación de cada una de las cajas de texto que manejen la obtención o captura de datos, estas pruebas serán correlacionadas con los estándares de manejo de seguridad en el campo informático ISO 27000 los que implica que el usuario deba mantener 12 caracteres como mínimo combinados entre mayúsculas y minúsculas números y caracteres especiales. En lo referente a las cajas de clave deben tener 8 caracteres combinaciones mayúsculas minúsculas y caracteres especiales.

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



El ingreso erróneo en cualquiera de estas dos cajas de captura de datos (usuario y clave) abortará el acceso al sistema y enviara a la página de acceso.

5.05.02Módulo Mantenimiento

El módulo de mantenimiento tiene que ser sometido a pruebas en las cuales se verifique el acceso idóneo de información hacia la base de datos; por esta razón inicialmente se determinará que los campos que manejen código de acceso en una tabla que tenga una función que me permita generar códigos secuenciales sin que estos se repitan y puedan generar errores de duplicidad de claves primarias., las cajas que requieran la captura de información deberán estar validadas de acuerdo al dato de ingreso ya sea esté números enteros o decimales, fecha y hora deberá ser los estandarizados por las normas ISO año 4 mes 2 día 2 separados mediante un -. En las cajas de texto que requiera identificar la cedula deberá estar validado que el número ingresado sea válido para poder continuar con las operaciones de ingreso.

5.05.03Pruebas de Interfaz de Usuario.

Esta prueba se realiza inicialmente verificando facilidad con la que el usuario se desenvuelve en realizar las distintas operaciones en el sistema de registro y control, así como la carga tiene que ser transparente para el usuario.

En este formulario verificamos que el salto de las cajas de texto se las realice con el enter, igualmente se cumple con la verificación de los estándares GUI que implica color de fondo de las cajas de texto, color de texto de la recuperación de datos

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



simetría en la distribución de cajas recuperadoras de información, diseño de ubicación de objetos de interfaz de usuario (text box, check box etc).

De igual manera se verifica el estándar de comunicación que existe entre los datos estableciendo tiempos de respuesta en la recuperación de información desde la base de datos hacia la aplicación y desde el aplicativo hacia la base de datos.

Por otro lado se determinó una prueba de ejecución del manejo de los iconos inicialmente que estén acorde con la información solicitada; en este punto se verifico ventanas y mensajes de alerta, ventanas y mensajes de información de la acción que se ha ejecutado, ventana y mensajes de captura de errores con el manejo de excepciones.

5.05.04 Pruebas de Desempeño.

Con la finalidad de poder ejecutar este tipo de pruebas y que las mismas tienen incidencia con la arquitectura montada, fue necesario estructurar el sistema de registro y control, en un ambiente distribuido que claramente se pueda evidenciar el servidor de datos, servidor de aplicaciones, y los clientes. La primera prueba realizada fue la verificación del tiempo de respuesta del cliente hacia el servidor de aplicaciones y posteriormente del servidor de aplicaciones al servidor de datos, el tiempo de respuesta final se lo determina con la suma total de tiempo realizado desde la petición de información hasta la recepción de la misma y esto dividido por dos. Se considera que el servidor debe estar separado para poder realizar las



pruebas, de esta manera podremos verificar el tiempo de respuesta de acceso remoto a la aplicación.

Continuando con las pruebas de desempeño tenemos que verificar tiempos válidos de respuesta para recibir y enviar información mediante el XML igualmente el tiempo de respuesta que se demora en generar este tipo de información.

5.05.05 Pruebas de Carga.

Para realizar esta prueba es necesario ejecutarla sobre una sola tabla la misma que es la que tiene más transaccionalidad dentro del proyecto la tabla "TBL_SOCIOS" con un soporte mínimo de 50 socios. Se medirá la persistencia de las tablas y de la programación para así obtener la persistencia de la transaccionalidad del sistema.

5.05.06 Pruebas de Validación.

El objetivo principal de las pruebas de validación es obtener información útil para la validación de la implementación. Una vez que el sistema de registro y control ha cumplido con la verificación de errores por lo tanto está libre de errores en tiempos de ejecución, lo que significa que está libre de errores lógicos.

El módulo de seguridad es donde se realiza la prueba de validación pues controla el acceso al sistema validando y verificando que los datos ingresados sean los correctos.



Se verifica que existan mensajes acorde a las acciones erróneas que se realizan en el ingreso de datos del sistema. La validación del software se la realiza con pruebas de verificación que demuestren que es 100% funcional

CAPITULO VI: Aspectos Administrativos

6.01 Recursos

Tabla 20

Recursos



RECURSOS HUMANOS			
HUMANO	NOMBRE	ACTIVIDAD	RESPONSABILIDAD
Tutor	Ing.	Director del proyecto	Responsabilidad de guiar e instruir
Director General		Autoriza la elaboración del sistema	Toma de decisiones
RECURSOS ECONÓMICOS			
DESCRIPCIÓN	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Computadora	2	620	1240
Impresora	1	200	200
Impresiones a color	500	0,25	125
Impresiones a b/n	800	0,05	40
Anillados	2	3	6
Empastados	1	6	6
Material de oficina	5	7	35
Internet	5	25,5	127,5
Llamadas telefónicas	28	1	28
Servicios Básicos	3	12,5	37,5
Capacitación	1	100	100
Tutorías	1	220	220
Seminario	1	500	500
Refrigerios	45	1,25	56,25
Transporte	100	0,25	25
Alimentación	100	2	200
	VALOR TOTAL		2946,25

6.02 Presupuesto

Tabla 21

Presupuesto

REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL



PRESUPUESTO			
DETALLE	VALOR UNITARIO	CANTIDAD	VALOR TOTAL
Computadora	2	620	1240
Impresora	1	200	200
Impresiones a color	500	0,25	125
Impresiones a b/n	800	0,05	40
Anillados	2	3	6
Empastados	1	6	6
Material de oficina	5	7	35
Internet	5	25,5	127,5
Servicios Básicos	3	12,5	37,5
Capacitación	1	100	100
Tutorías	1	200	200
Seminario	1	520	520
VARIOS			
Refrigerios	45	1,25	56,25
Transporte	100	0,25	25
Llamadas telefónicas	28	1	28
Alimentación	100	2	200
VALOR TOTAL			2946,25

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



6.03 Cronograma

El cronograma es donde se muestra la definición del tiempo utilizado para el desarrollo del proyecto en una forma continua con las diferentes actividades a realizar.

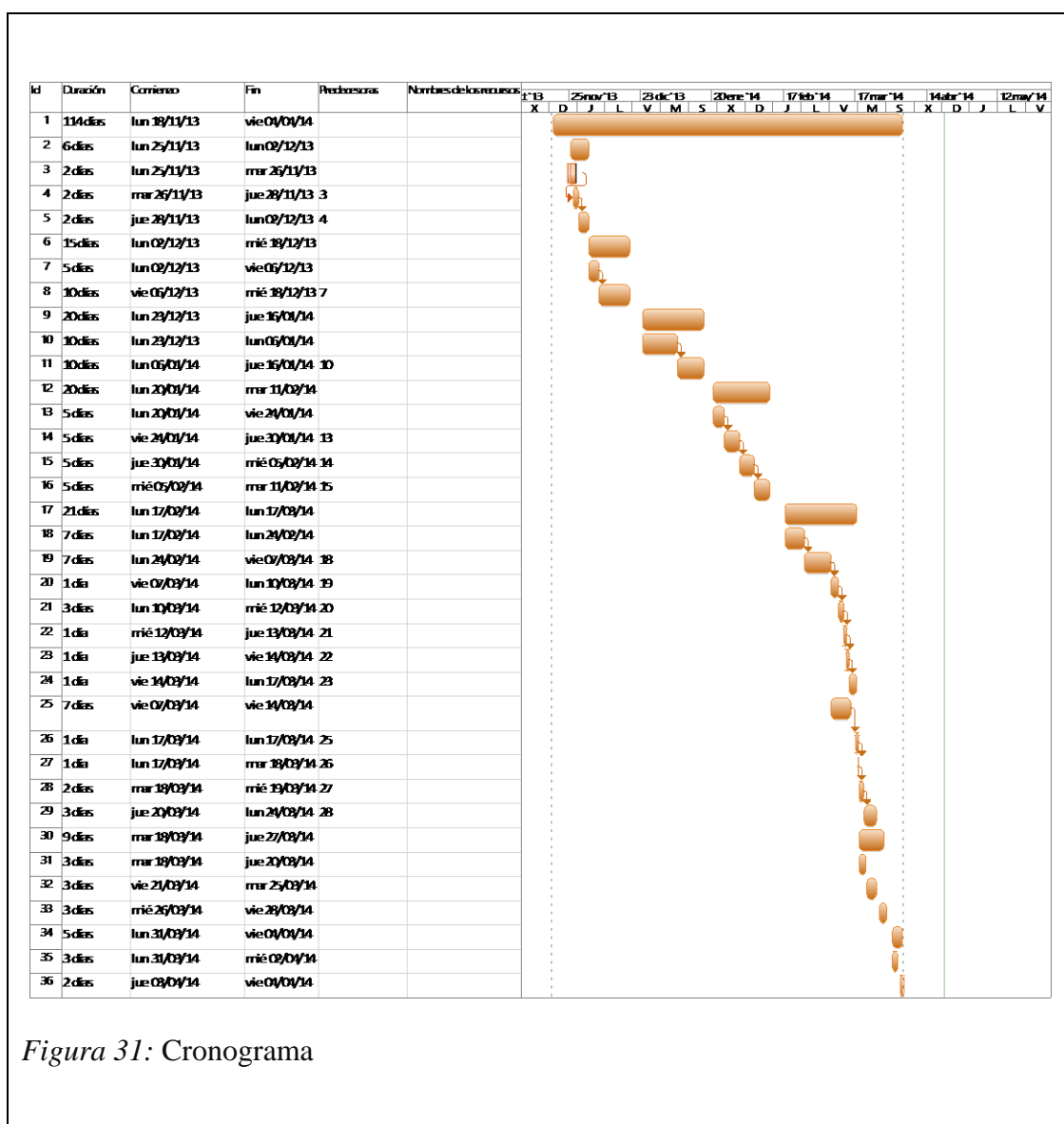


Figura 31: Cronograma



CAPITULO VII: Conclusiones y Recomendaciones

7.01 Conclusiones

Llegamos a la conclusión que la investigación realizada ayudará en lo siguiente:

- Con la implementación del sistema de registro y control de pago de alcúotas tendremos la optimización de procesos operacionales a través de los diferentes módulos que tiene el sistema mejorando así la información financiera.
- El sistema ayudará a llevar el registro de los socios, informes de los pagos realizados de manera automatizada sin correr el riesgo de perder la información.
- El sistema permitirá generar actas de las asambleas realizadas
- Se generará un comprobante por los pagos realizados por los socios.
- Los informes que se realizan pondrán imprimirse ya que se generaran en formato PDF.

7.02 Recomendaciones

Para que se pueda cumplir los requerimientos del barrio se recomienda:

- Capacitar al personal responsable del manejo del sistema sobre el programa con el fin de que utilicen y conozcan adecuadamente el funcionamiento del software.



- Generar reportes constantemente de los procesos que se realizan internamente
- Es muy importante que se realicen en un periodo continuo los respaldos respectivos de la información existente.
- Brindar el mantenimiento respectivo al sistema donde la información guardada sea la correcta.
- En caso de alguna falla o inconveniente en el funcionamiento de la aplicación consultar en el presente documento.



ANEXOS

***REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL***



ANEXO 01

Resumen Narrativo de los Objetivos	Indicadores	Medios de Verificación	Supuestos
FIN Eficiencia en el registro y control de pagos e información personal	Menor porcentaje de error en el cuadro de cuentas	Con el cuestionario realizado al presidente del barrio, el resultado indica que sería de gran ayuda la aplicación para el registro y control de información y pago de alcúotas, el cual generara reportes donde se podrá ver el rendimiento del barrio	<ul style="list-style-type: none"> No tener la infraestructura para la instalación del sistema. No disponer de espacio físico. Tener equipos que no funcionen. Cambio de opinión en el barrio y negar la implementación de la aplicación.
PROPOSITO Procesos financieros adecuados	<ol style="list-style-type: none"> Mayor número de socios atendidos al mes. Elaboración de un manual de funciones para los puestos laborales desde el año 2013 hasta el 2014 + será en la utilización de una aplicación para el control de procesos. <p>En el año 2014 se agilizará el control de procesos e información que en el año 2013 no fueron solucionados.</p>	Realizar el seguimiento del proyecto tomando en cuenta los reportes diarios.	<ul style="list-style-type: none"> El barrio pida módulos no acordados antes. El barrio no proporcione la información requerida para la aplicación. El presidente del barrio no quiera la implementación de la aplicación informática.
COMPONENTES 1. Apoyo con herramientas tecnológicas para la	<ol style="list-style-type: none"> En el año 2013 no existe integridad de información a ser 	<ul style="list-style-type: none"> En la actualidad existe una demanda de aplicaciones 	<ul style="list-style-type: none"> No tener garantía de los equipos en caso de daño. Suban los costos de las

ALICUOTAS MEDIANTE UN SISTEMA INFORMÁTICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL




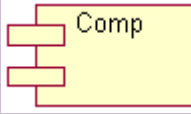
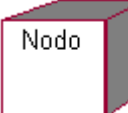
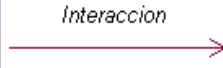
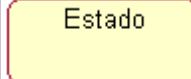


<p>automatización de procesos. El personal está muy bien capacitado para el manejo de los procesos.</p> <p>2. Procesos e información bien definidos.</p> <p>3. Se dispone de un sistema tecnológico el cual ayuda al control de procesos e información en el barrio Santa Isabel.</p>	<p>procesada, para el año 2014 se puede automatizar el control de los pagos</p> <p>2. En el año 2013 se ha sabido que existe un bajo interés por parte del tesorero del barrio en la generación de información adecuada.</p> <p>3. Desde el año 2012 se ha detectado pérdida de información por el mal manejo de procesos, el cual en el 2014 se desea solucionar con una aplicación informática.</p>	<p>s informática s que ayudan al control y se los encuentra en internet.</p> <ul style="list-style-type: none"> • En el país existen dependencias tecnológicas extranjeras por la cual se puede adquirir los dispositivos necesarios. 	<p>herramientas tecnológicas.</p>
<p>ACTIVIDADES</p> <p>1.1 Minimizar tiempo en el control de procesos</p> <p>1.2 Facilitar el registro de la información</p> <p>1.3 Organizar las actividades de trabajo</p> <p>1.4 Facilitar la generación de</p>	<p>Materiales</p> <p>Equipo de cómputo 650</p> <p>Hojas e impresiones 500</p> <p>Otros costos</p> <p>Transporte 100</p> <p>Llamadas 50</p> <p>Reserva</p> <p>200</p>	<ul style="list-style-type: none"> • Verificar módulo de seguridad • Verificar módulo de mantenimiento • Verificar módulo de reglas de negocio • Verificar modulo de 	<ul style="list-style-type: none"> • Beneficiarios directos estén satisfechos con la aplicación proporcionada. • Seguridad y rapidez en el control de procesos, información pagos.

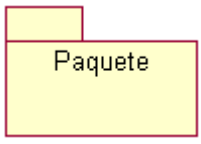
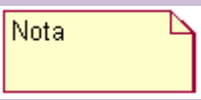
reportes. 1.5 Evitar la pérdida de documentos. 1.6 Automatizar los procesos operacionales 1.7 Evitar la duplicación de documentos con información incorrecta.	TOTAL PRESUPUESTO 1500	procesos • Verificar módulo de reportes.	• Personal capacitado para el manejo de procesos. • Incremento de ingresos para el barrio.
--	---	---	---

ANEXO 02

ELEMENTOS	Clase		Describe un conjunto de objetos que comparten los mismos atributos, métodos, relaciones y semántica. Las clases implementan una o más interfaces.
	Clase activa		Se trata de una clase, en la que existen procesos o hilos de ejecución concurrentes con otros elementos. Las líneas del contorno son más gruesas que en la clase "normal"
			Agrupación de métodos u operaciones que especifican un servicio de una clase o componente, describiendo su comportamiento, completo o parcial, externamente visible. UML permite emplear un círculo para representar las interfaces, aunque lo más normal es

E S T R U C T U R A L E S	Interfaz	 Interfaz	emplear la clase con el nombre en cursiva.
	Colaboración	 Colaboracion	Define una interacción entre elementos que cooperan para proporcionar un comportamiento mayor que la suma de los comportamientos de sus elementos.
	Caso de uso	 Caso de Uso	Describe un conjunto de secuencias de acciones que un sistema ejecuta, para producir un resultado observable de interés. Se emplea para estructurar los aspectos de comportamiento de un modelo.
	Componente	 Comp	Parte física y por tanto reemplazable de un modelo, que agrupa un conjunto de interfaces, archivos de código fuente, clases, colaboraciones y proporciona la implementación de dichos elementos.
	Nodo	 Nodo	Elemento físico que existe en tiempo de ejecución y representa un recurso computacional con capacidad de procesar.
	Interacción	 Interaccion	Comprende un conjunto de mensajes que se intercambian entre un conjunto de objetos, para cumplir un objetivo específico.
	Máquinas de	 Estado	Especifica la secuencia de estados por los que pasa un objeto o una interacción, en



Elementos de agrupación	estados		respuesta a eventos.
	Paquete		Se emplea para organizar otros elementos en grupos.
Elementos de notación	Nota		Partes explicativa de UML, que puede describir textualmente cualquier aspecto del modelo

8 Instalación de programas utilizados

Instalación de Visual Studio 2010

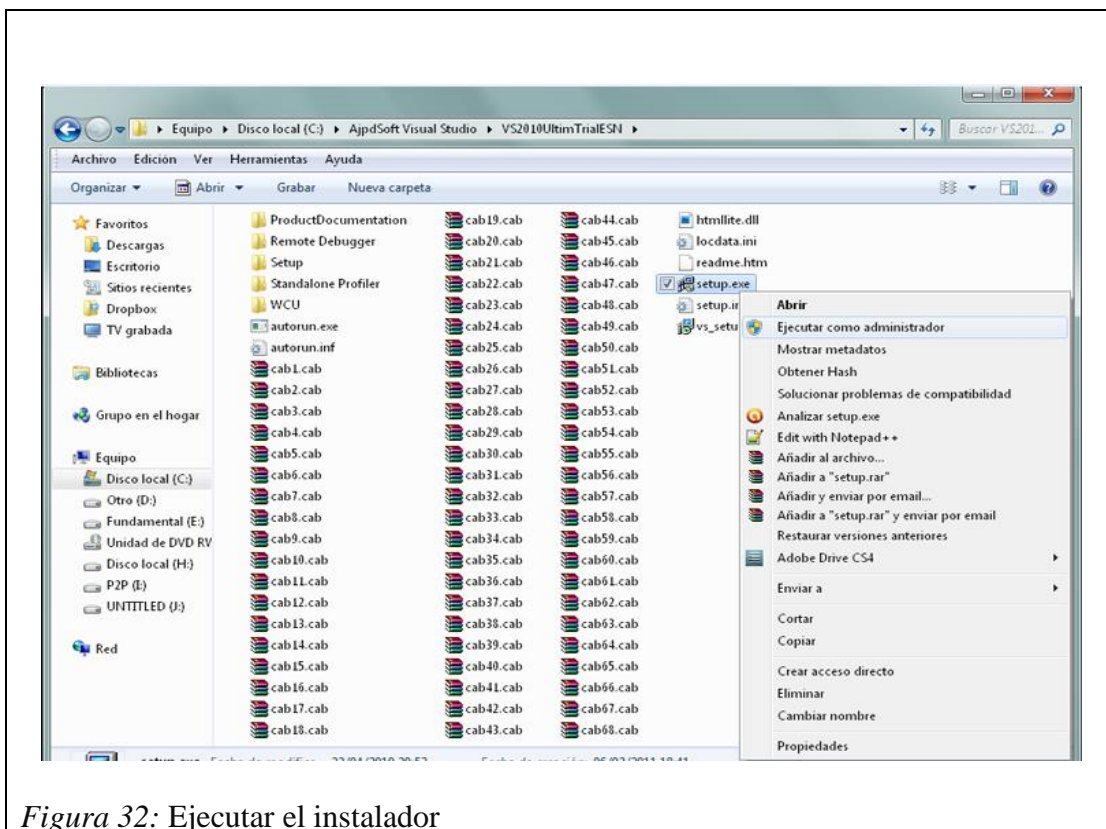


Figura 32: Ejecutar el instalador

Para instalar debemos hacer clic derecho sobre el icono y seleccionar “Ejecutar como administrador”.



Figura 33: Iniciar la instalación

Se iniciara el asistente para la instalación del Visual Studio 2010, pulsamos en “Instalar Microsoft Visual Studio 2010”.

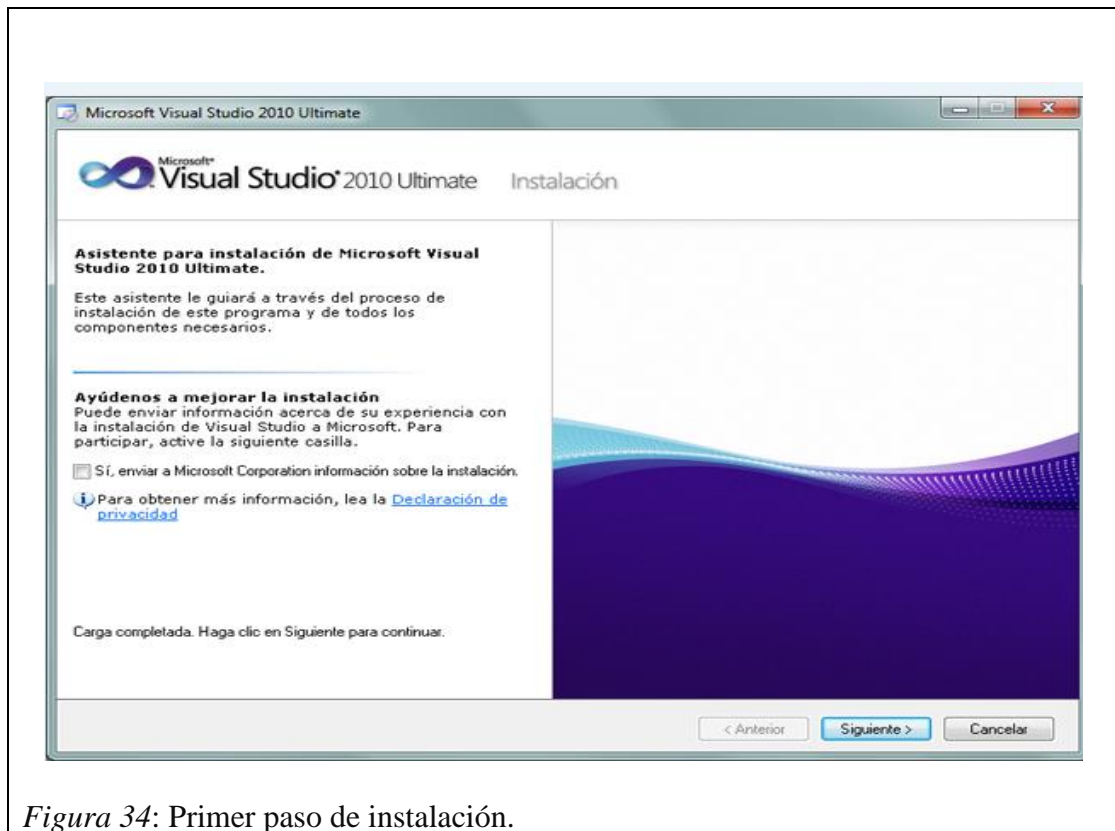


Figura 34: Primer paso de instalación.

En el primer paso de la instalación desmarcamos “Si, enviar a Microsoft Corporation información sobre la instalación” (si no queremos enviar esta información) y pulsamos “Siguiente”.

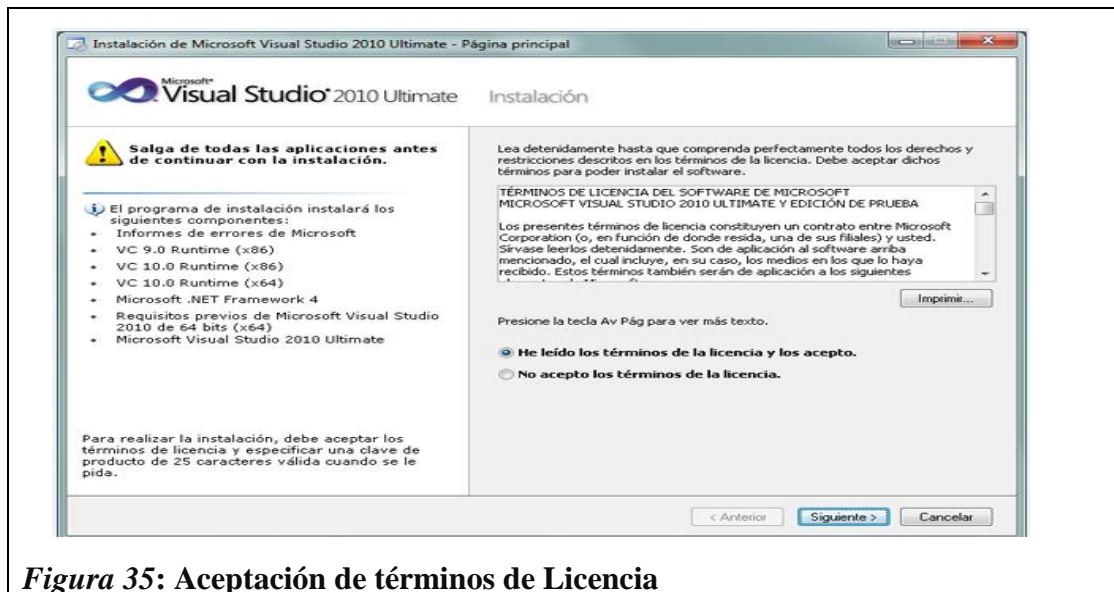


Figura 35: Aceptación de términos de Licencia

Leer los términos de licencia del software, si estamos de acuerdo marcaremos “he leído los términos de licencia y los acepto”. Pulsaremos “Siguiente” para continuar.

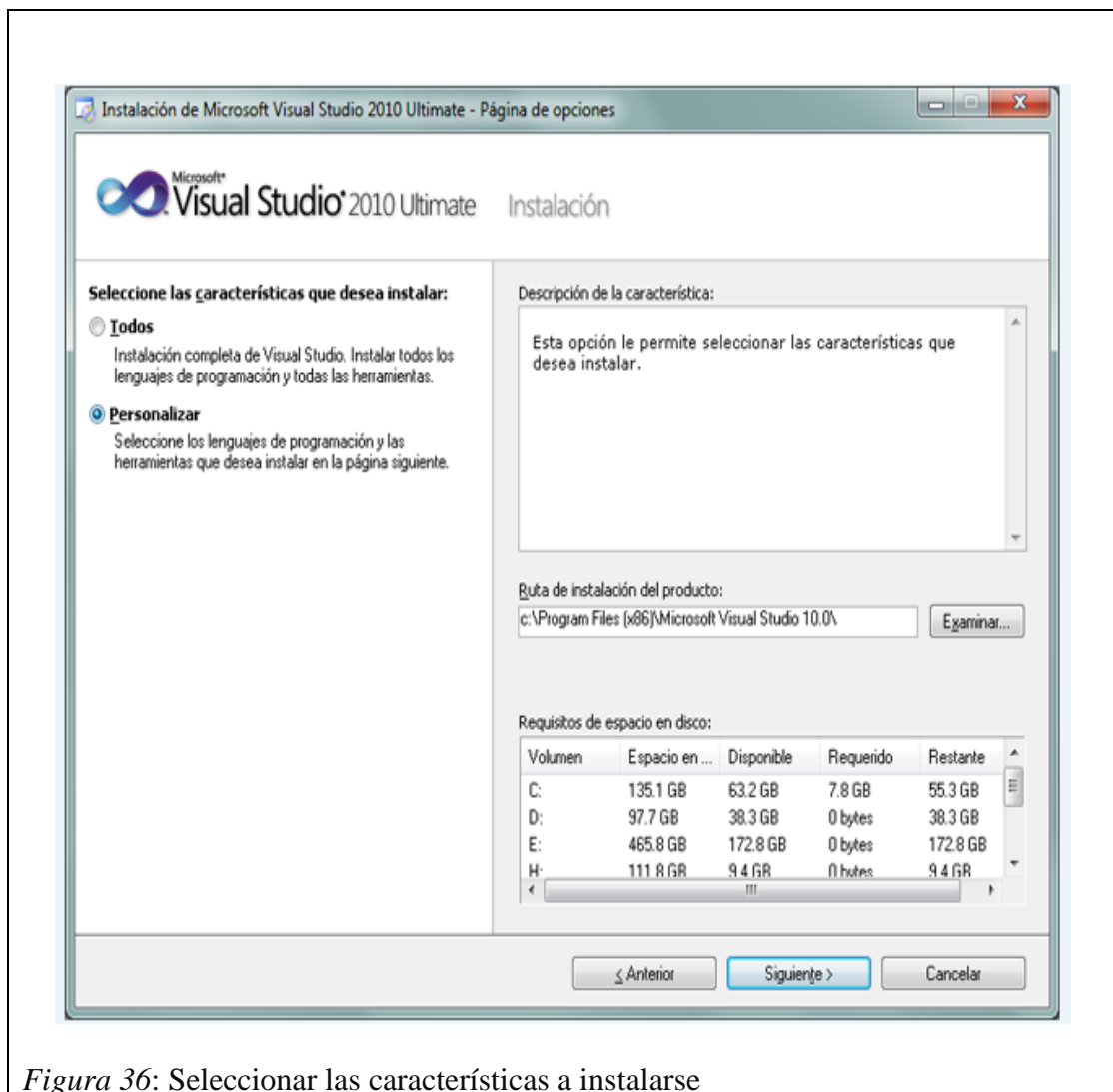


Figura 36: Seleccionar las características a instalarse

A continuación marcaremos “Personalizar” para seleccionar los lenguajes de programación y las herramientas que se quieran instalar. En “Ruta de instalación del producto” indicaremos la unidad y carpeta de destino de la instalación.

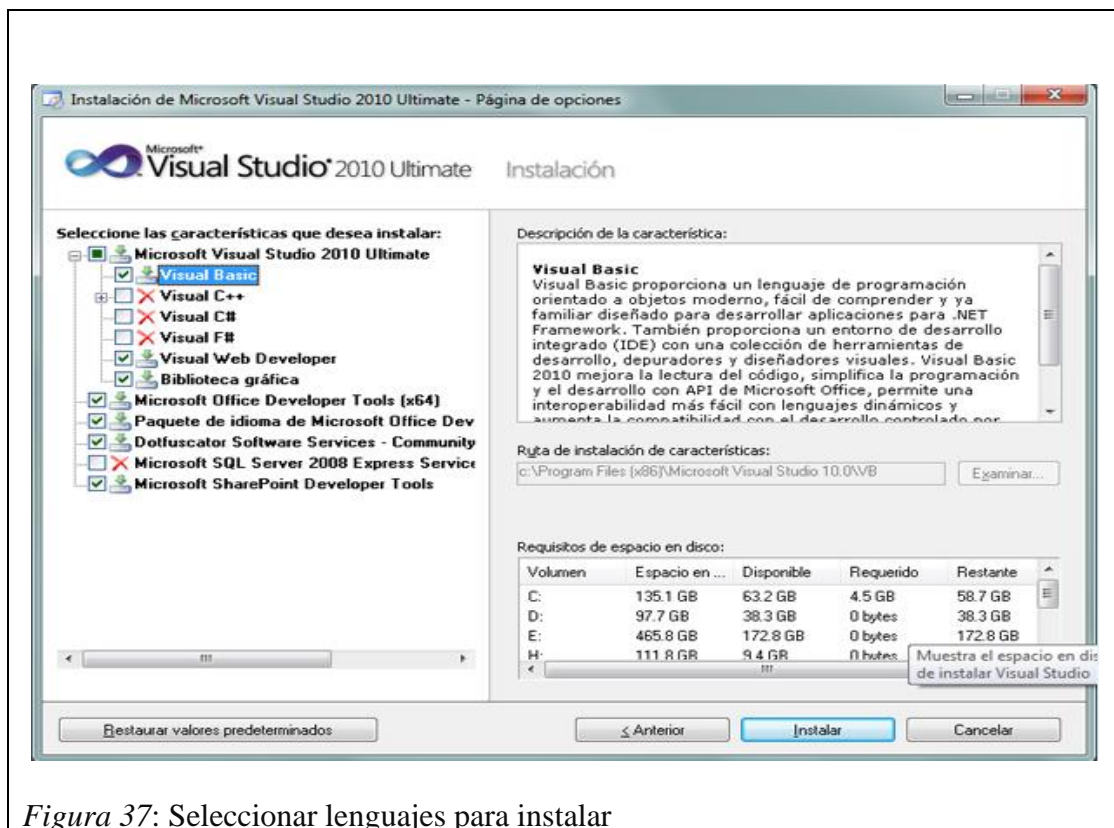


Figura 37: Seleccionar lenguajes para instalar

Seleccionamos los items a instalar:

- Visual Basic.
- Visual C++.
- Visual C#.
- Visual F#.
- Microsoft Office Developer Tools.
- Dotfuscator Software Services.
- Microsoft SQL Server 2008 Express.
- Microsoft SharePoint Developer Tools.

Una vez seleccionadas las características a instalar pulsaremos en el botón "Instalar":

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

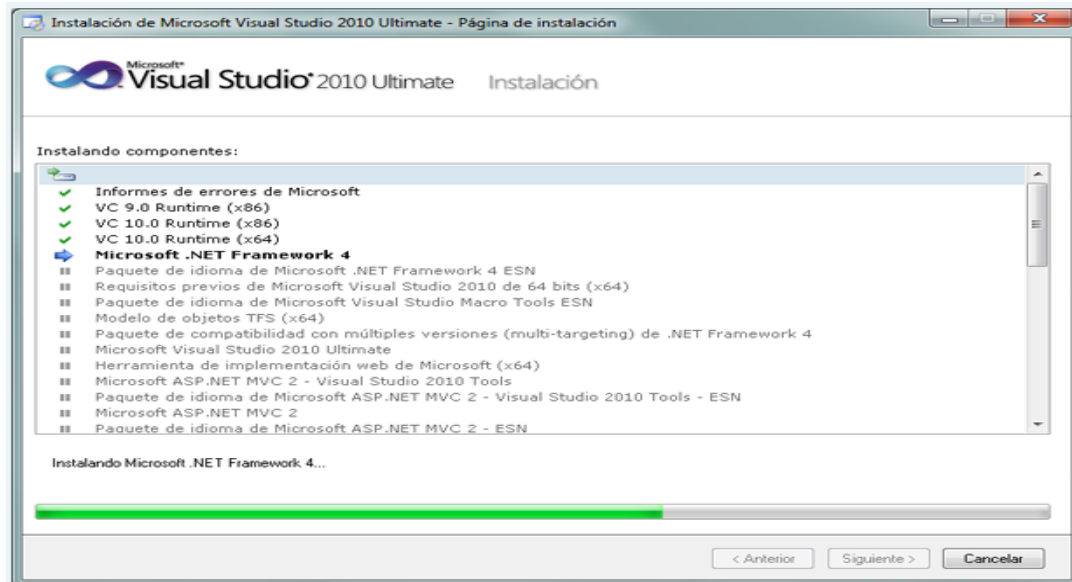


Figura 38: Espera de Instalación

Se iniciará la instalación de Microsoft Visual Studio 2010 Ultimate.



Figura 39: Finalización de instalación

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

Tras la instalación, el asistente para instalar Visual Studio mostrará la ventana, con el texto "Correcto. Se instaló Visual Studio 2010 y finalizó el programa de instalación". Pulsaremos "Finalizar".



Figura 40: Reinicio del computador después de la instalación

Tras el proceso de instalación puede que pida reiniciar el equipo.

Con el texto: "Debe reiniciar el equipo para completar la instalación. El programa de instalación continuará automáticamente después de haber reiniciado el equipo".

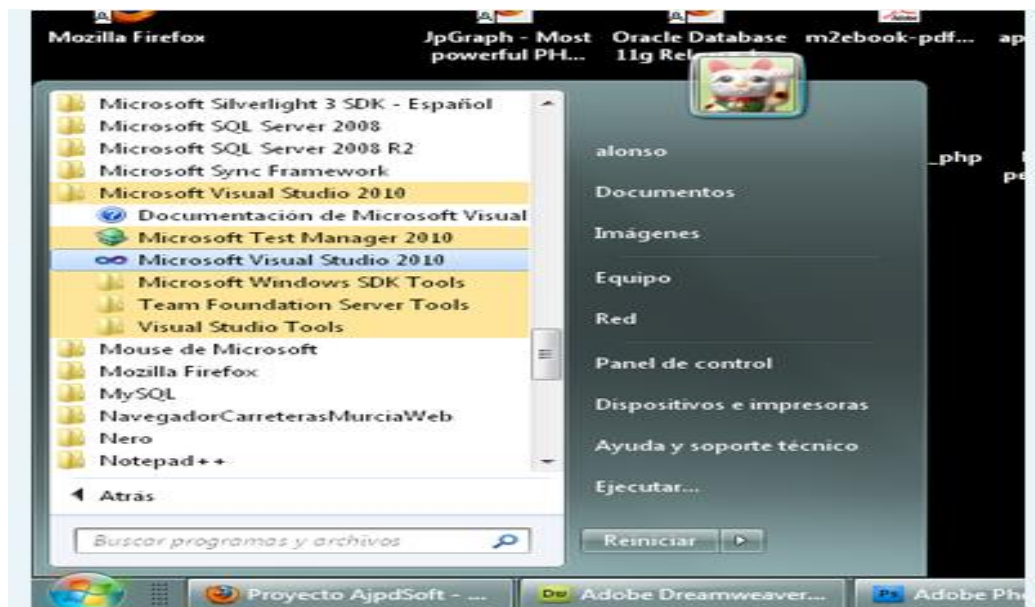


Figura 41: Iniciar Visual Studio.

Desde el botón "Iniciar" - "Todos los programas" podremos ver el grupo de programas "Microsoft Visual Studio 2010".

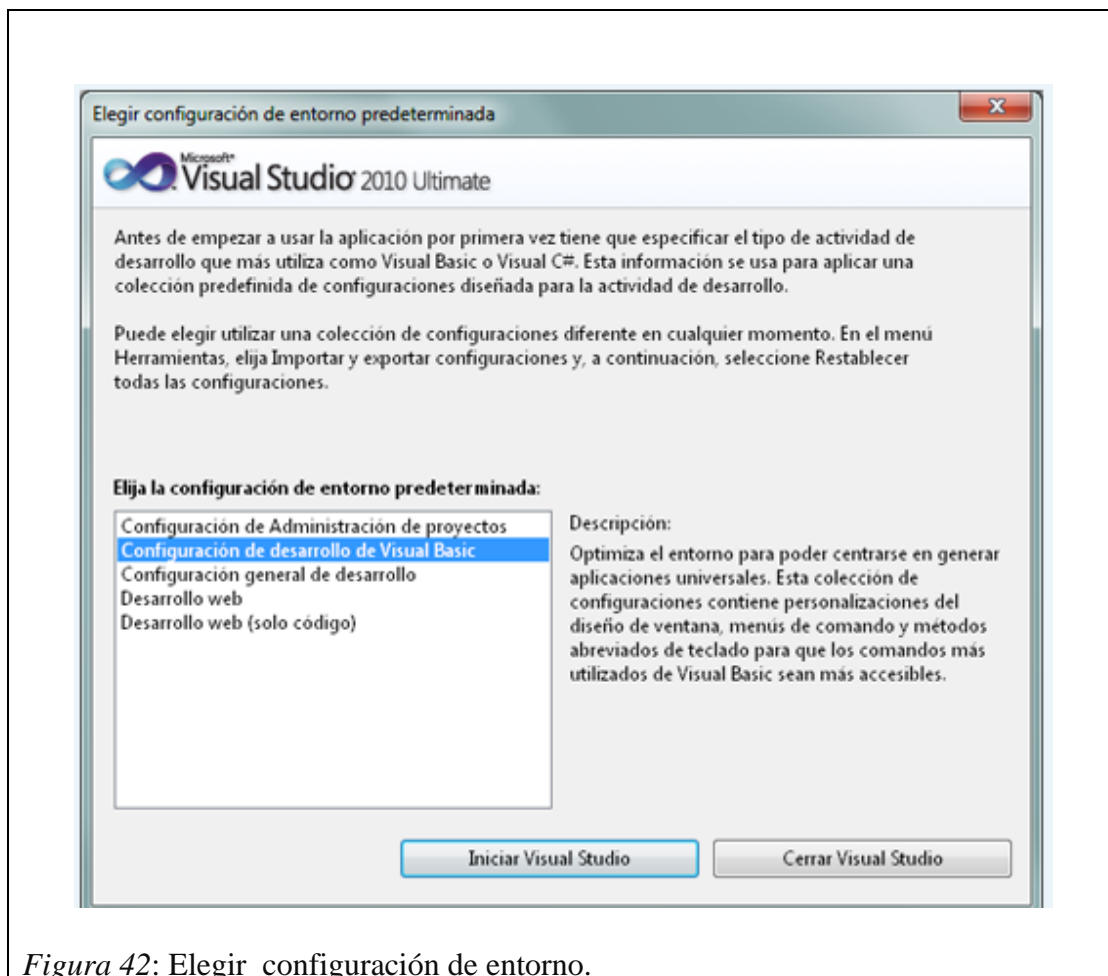


Figura 42: Elegir configuración de entorno.

Tras iniciar por primera vez Microsoft Visual Studio 2010 Ultimate, seleccionaremos la configuración de entorno predeterminada, por ejemplo "Configuración de desarrollo de Visual Basic".

Seleccionamos "Iniciar Visual Studio".

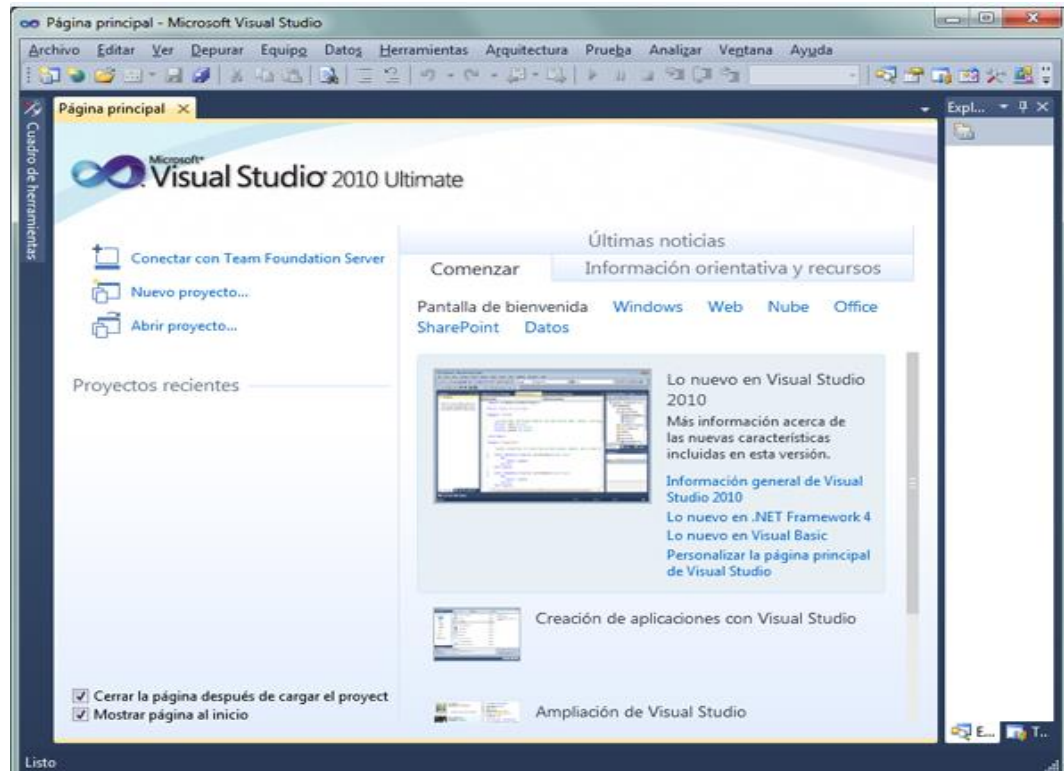


Figura 43: Finalización de la instalación

Ya podremos disponer de este entorno de programación.

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

INSTALACION DE SQL SERVER 2008

1. Primero se ejecuta la instalación de SQL SERVER 2008 desde el CD, luego nos sale una ventana que es el asistente de la Instalación, hacemos clic en la Pestaña "Instalación" y luego en la opción "Nueva Instalación independiente de SQL Server o agregar características a una instalación existente."

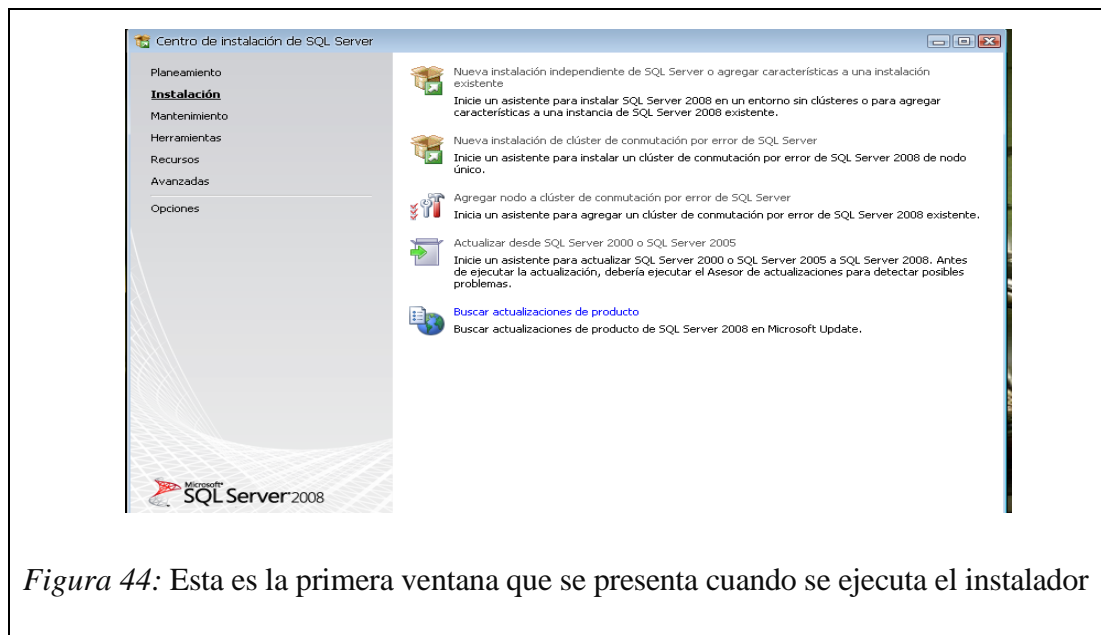
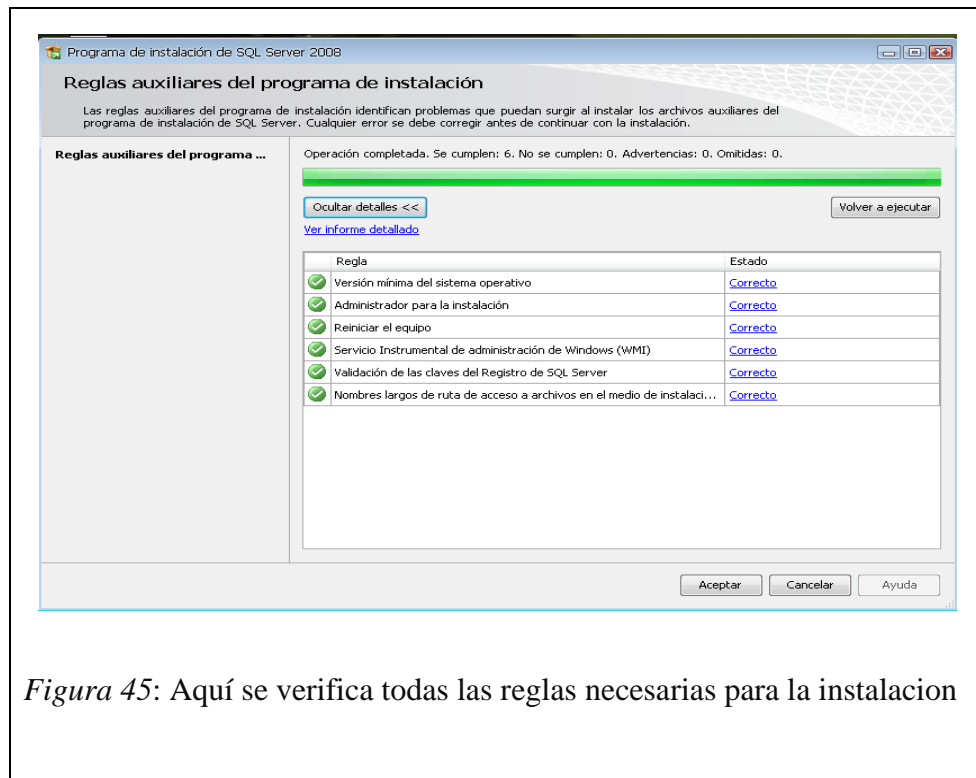
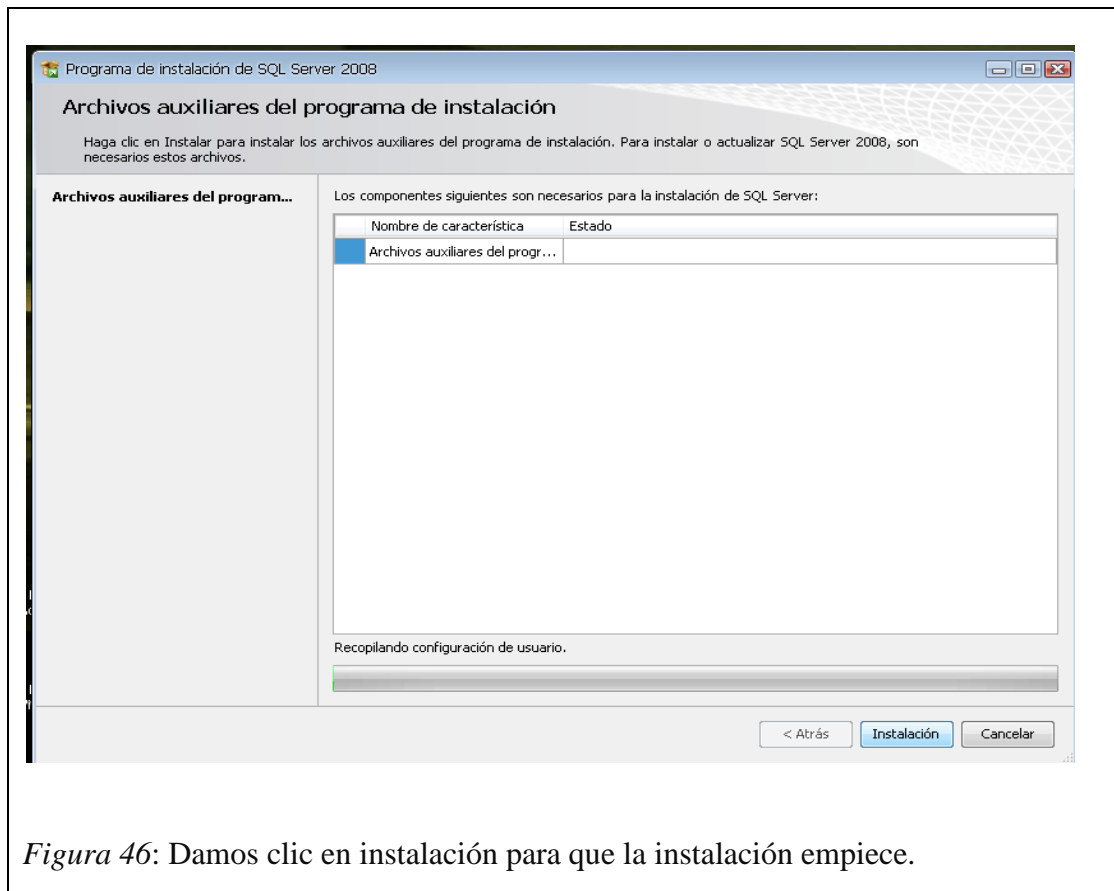


Figura 44: Esta es la primera ventana que se presenta cuando se ejecuta el instalador

2. En la siguiente ventana comienza a cargar algunas Reglas auxiliares que son necesarias para la instalación de SQL Server 2008, si todos están en Estado **Correcto** hacer clic en el botón siguiente de lo contrario si no están algunos en Estado **Correcto** no se puede seguir o continuar con la instalación (para ver las reglas clic en el botón mostrar detalles).



3. En la siguiente ventana solo hacemos clic en el botón instalación para instalar algunos archivos auxiliares del programa de Instalación.



4. Luego de haber instalado los archivos auxiliares que son necesarios para la Instalación de SQL Server 2008, la siguiente ventana que nos sale es la clave del producto y nos dan dos opciones: "Especifique una edición gratuita" y "Escriba la clave del Producto", por defecto sale seleccionado la segunda opción y la clave, así que solo damos clic en el botón siguiente.

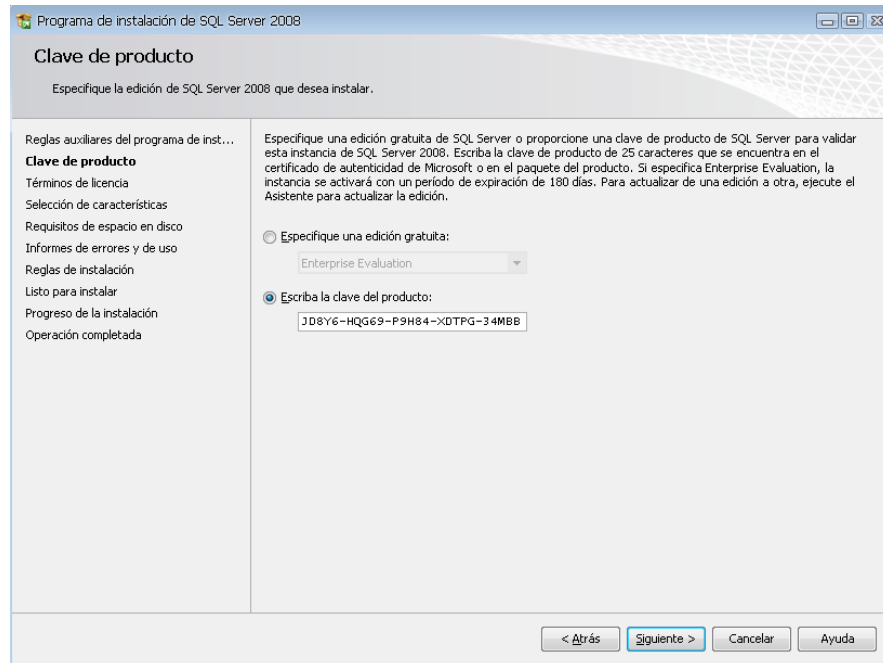


Figura 47: En esta ventana solo damos clic en siguiente ya que por defecto viene la clave del producto.

5. En esta ventana es si sobre los términos de la licencia del producto, hacemos clic en la opción “Acepto los términos de licencia” y luego clic en Siguiente.

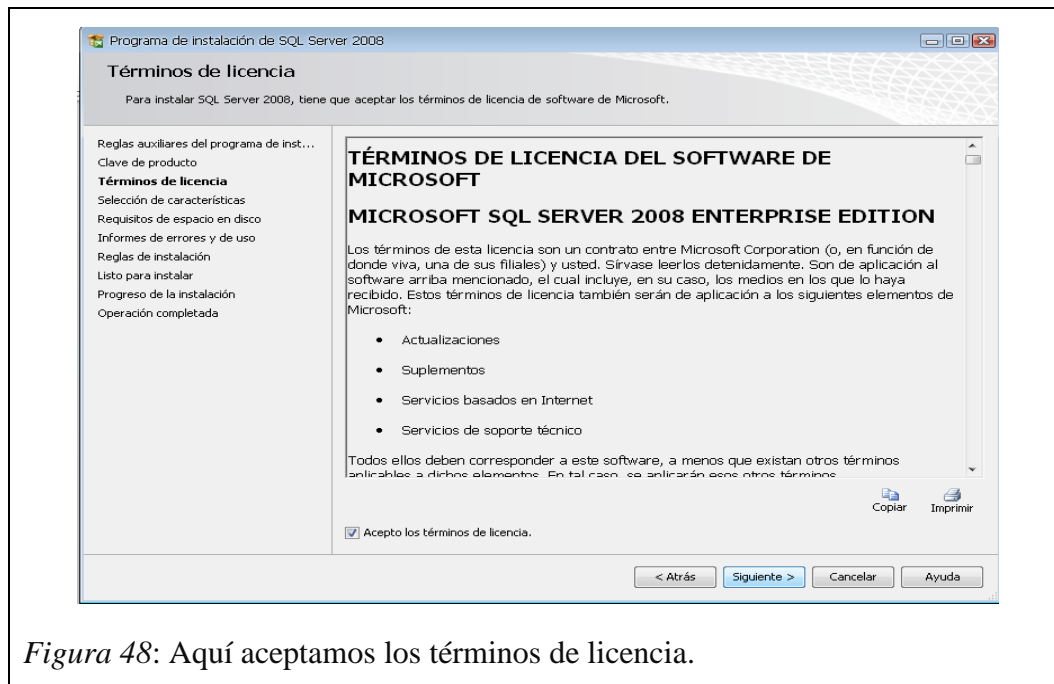


Figura 48: Aquí aceptamos los términos de licencia.

6. La siguiente ventana que nos muestra son las características que queremos agregar a la instalación, lo recomendable es seleccionar todas las características y luego clic en Siguiente.

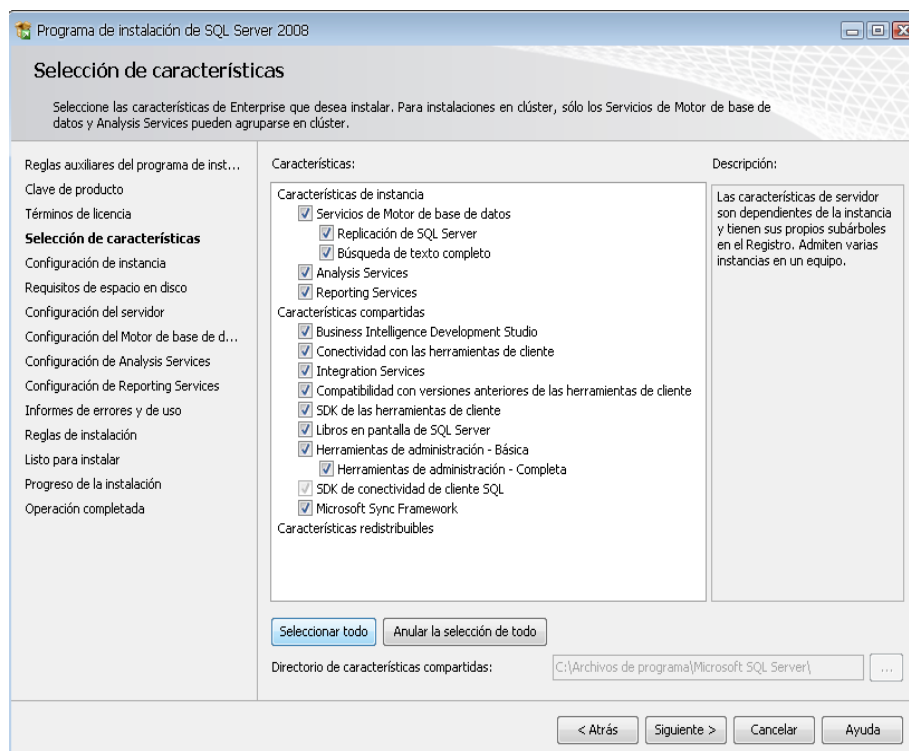


Figura 49: Aquí seleccionamos las características que se agregarán a la instalación.

7. Luego de haber agregado las características, la siguiente ventana a mostrar es sobre la configuración de la instancia de SQL Server 2008; es decir el nombre de la instalación, su ID y su ubicación en donde se instalará el producto. Nos dan 2 opciones: "Instancia predeterminada" y "Instancia con nombre", seleccionar la primera opción y luego clic en siguiente. En la siguiente ventana también damos clic en siguiente.



Programa de instalación de SQL Server 2008

Configuración de instancia

Especifique el nombre y el id. de instancia de SQL Server.

Reglas auxiliares del programa de inst...
Clave de producto
Términos de licencia
Selección de características
Configuración de instancia
Requisitos de espacio en disco
Configuración del servidor
Configuración del Motor de base de d...
Configuración de Analysis Services
Configuración de Reporting Services
Informes de errores y de uso
Reglas de instalación
Listo para instalar
Progreso de la instalación
Operación completada

☒ Instancia predeterminada
☐ Instancia con nombre: MSSQLSERVER

Id. de instancia: MSSQLSERVER

Directorio raíz de instancia: C:\Archivos de programa\Microsoft SQL Server\

Directorio de SQL Server: C:\Archivos de programa\Microsoft SQL Server\MSSQL10.MSSQLSERVER
Directorio de Analysis Services: C:\Archivos de programa\Microsoft SQL Server\MSAS10.MSSQLSERVER
Directorio de Reporting Services: C:\Archivos de programa\Microsoft SQL Server\MSRS10.MSSQLSERVER

Instancias instaladas:

Instancia	Características	Edición	Versión	Id. de instancia
-----------	-----------------	---------	---------	------------------

< Atrás Siguiendo > Cancelar Ayuda

Figura 50: aquí configuramos la instancia, el ID y la dirección donde se realizara la instalación.

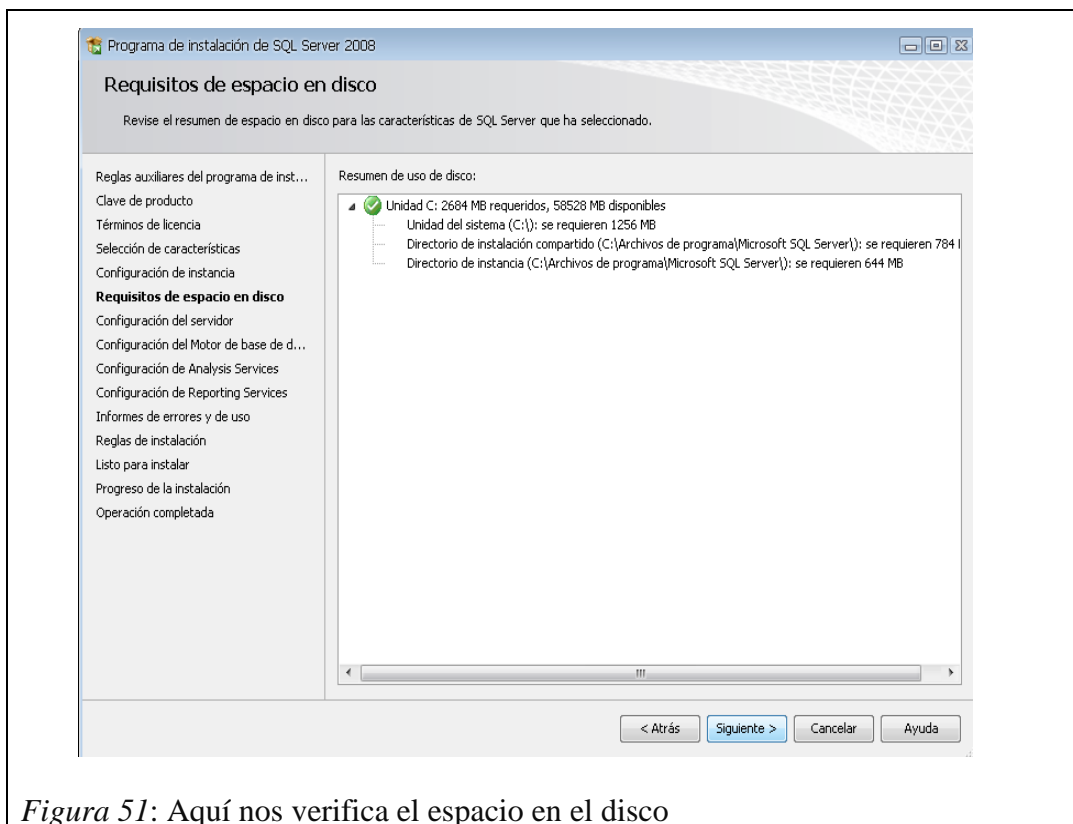


Figura 51: Aquí nos verifica el espacio en el disco

- En la siguiente ventana configuramos el servidor y para eso nos pide una cuenta de usuario, lo cual podemos crear un usuario específico para la instalación SQL, o de lo contrario podemos usar el usuario actual. En este caso usaremos "NT AUTHORITY\Servicio de red", luego damos clic en la pestaña "INTERCALACION".

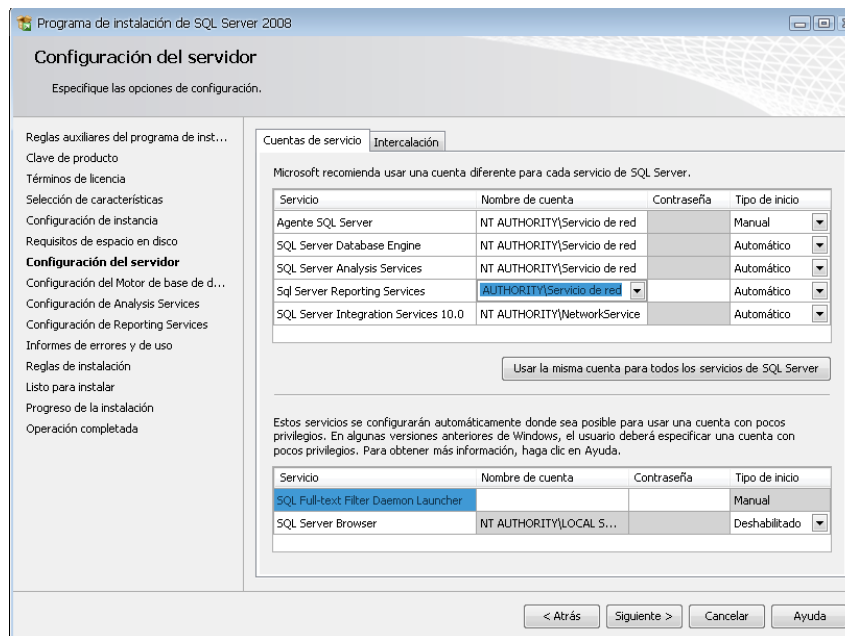


Figura 52: En esta ventana configuramos el servidor.

9. En esta pestaña de Intercalación configuramos el “motor de Base de Datos” y el “Servicio de Análisis”, para lo cual hacemos clic en el botón Personalizar en cada uno. Luego lo personalizamos como nos muestrafiguras “Motor de base de datos” y “AnalysisServices” y luego hacemos clic en siguiente.

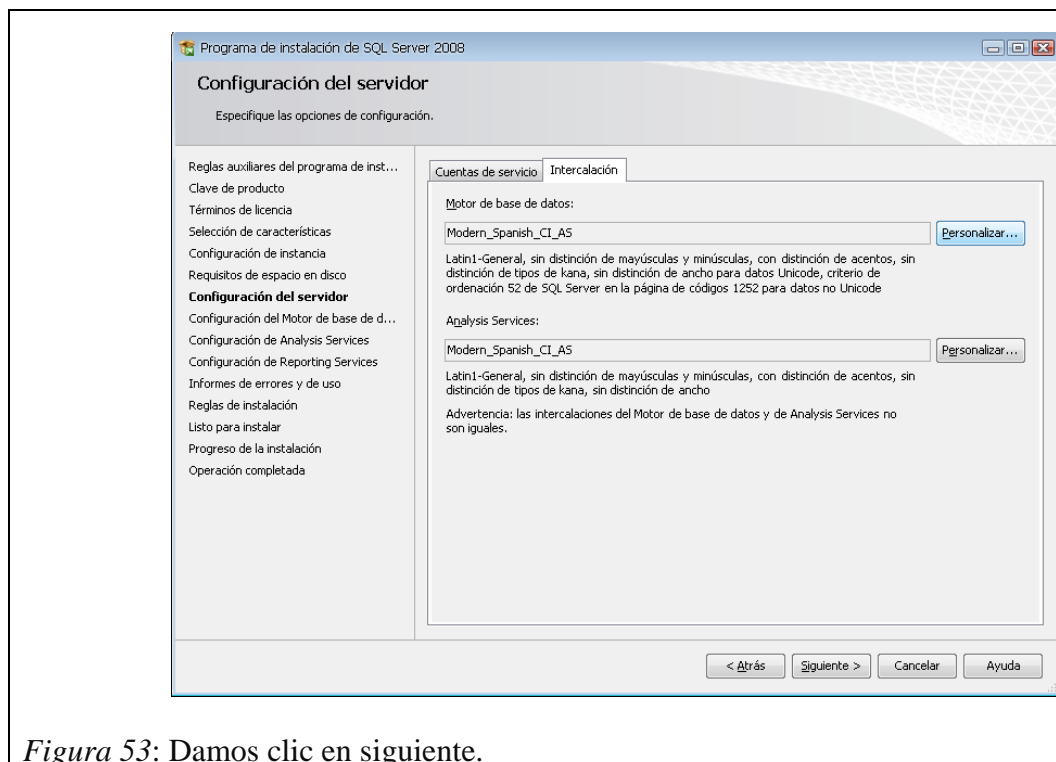


Figura 53: Damos clic en siguiente.

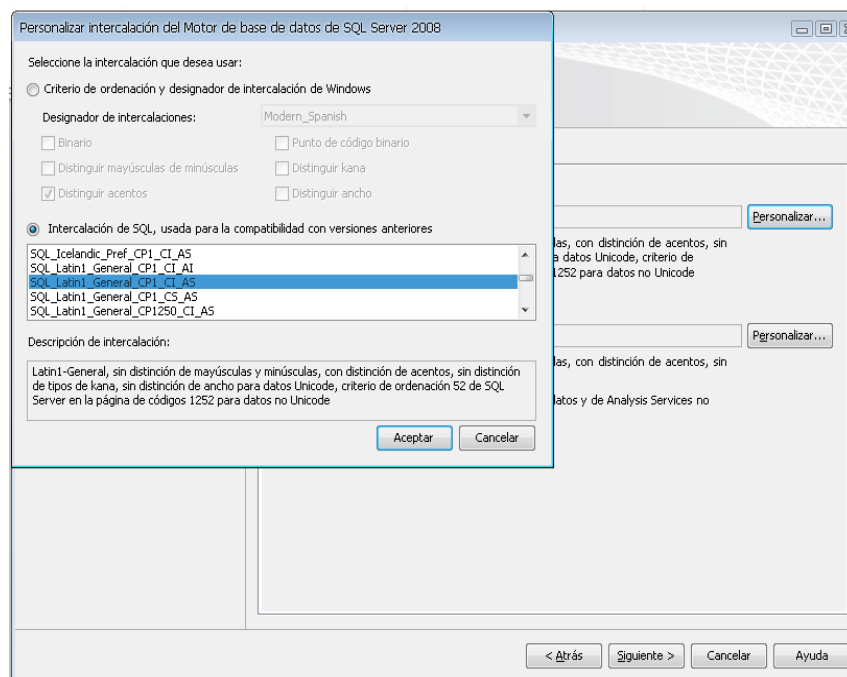


Figura 54: Figura Motor de Base de Datos

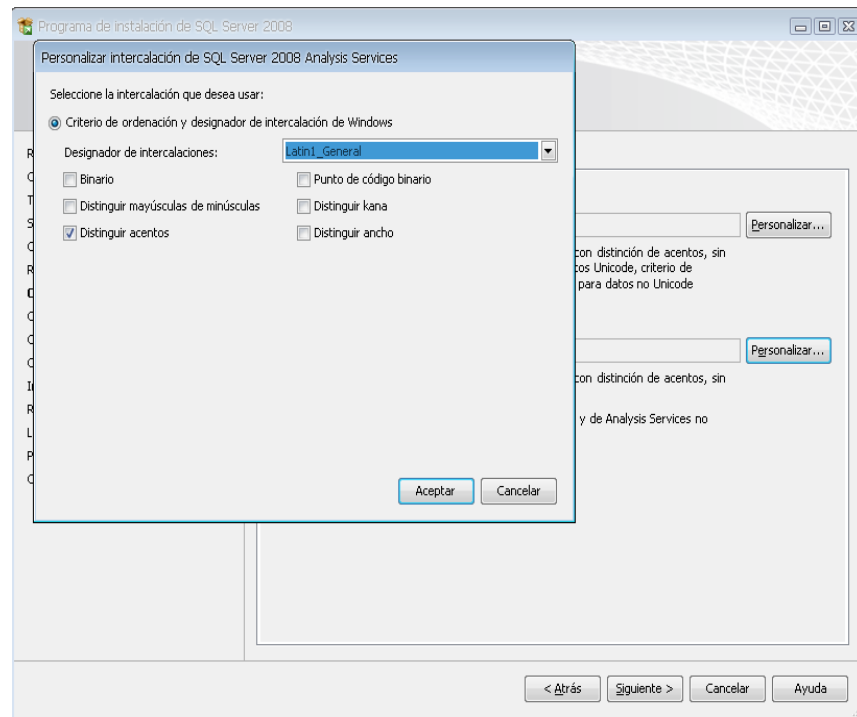


Figura 55: Figura Analysis Services

10. Luego la siguiente ventana que nos muestra es la configuración del MOTOR DE BASE DE DATOS, nos dan 2 opciones MODO AUTENTICACION DE WINDOWS Y MODO MIXTO (autenticación de SQL Server y de Windows). Lo recomendable es seleccionar el MODO MIXTO, al seleccionar esta opción se activan las cajas de contraseña para el inicio de sesión de autenticación de SQL Server (inicio de sesión de sa). Luego nos pide que agreguemos los administradores de SQL Server para lo cual agregamos el usuario actual; luego clic en siguiente. La siguiente ventana a mostrar también agregamos el usuario actual y luego siguiente.

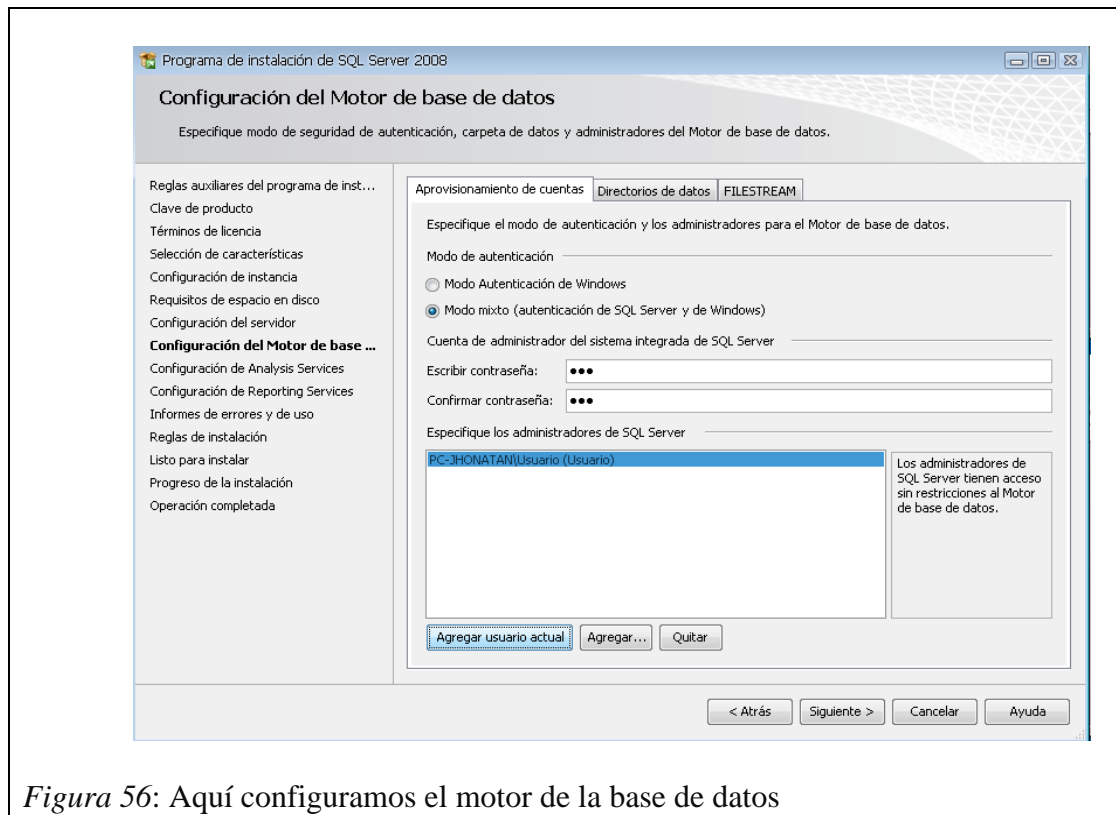
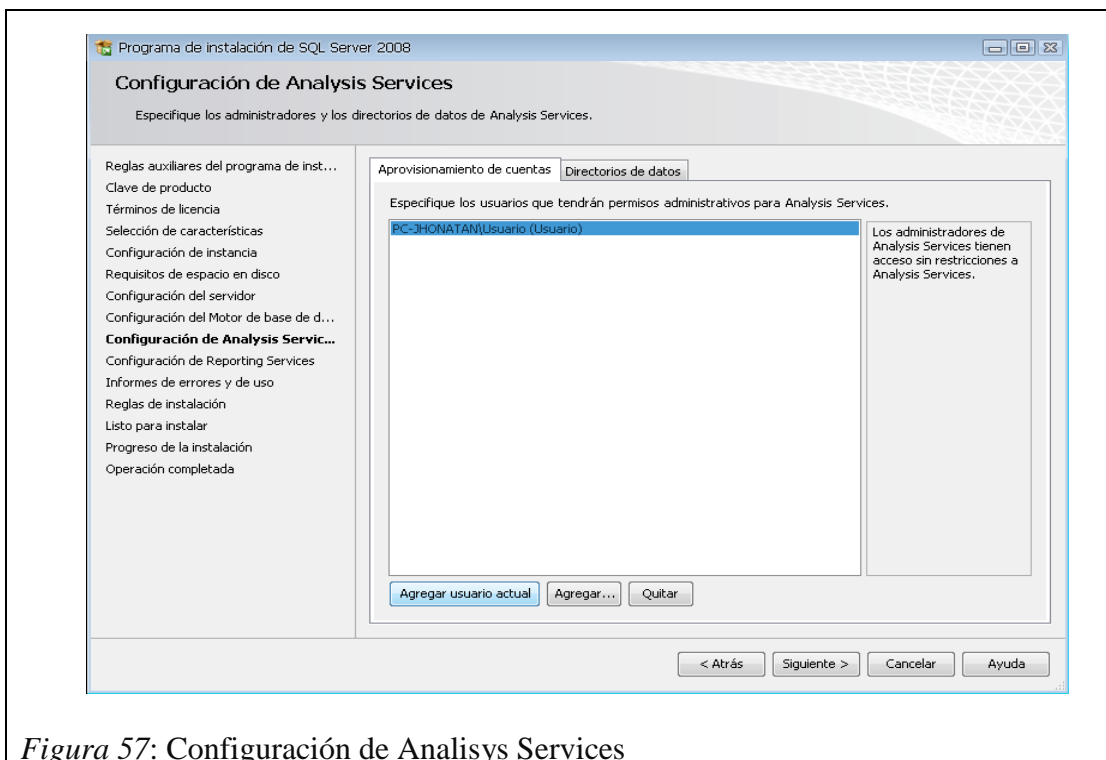


Figura 56: Aquí configuramos el motor de la base de datos



11. En esta ventana de Configuración de ReportingServices también nos dan 3 opciones, por defecto sale seleccionado la primera opción, así que solo damos clic en siguiente (2 veces).

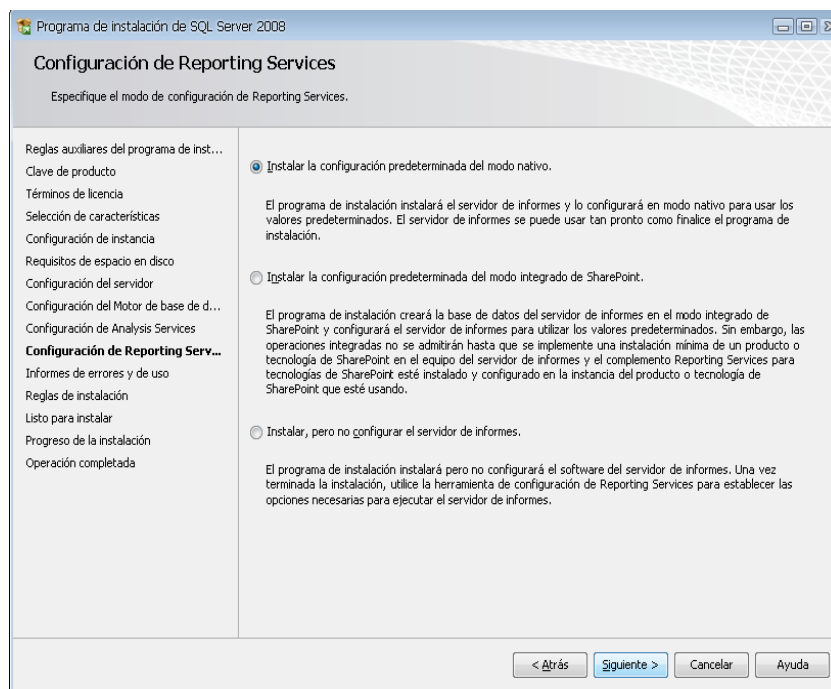


Figura 58: Configuración de Reporting Services

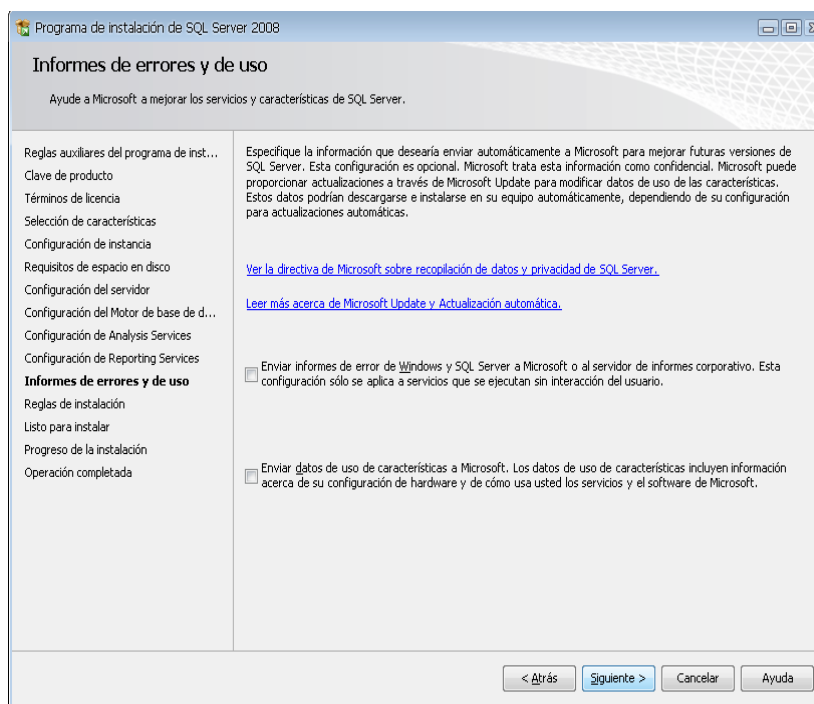


Figura 59: Esta ventana nos indicara un informe de los errores encontrados

12. Luego la siguiente ventana a mostrar son reglas que nos piden para poder **INSTALAR SQL SERVER**. Si todas las reglas están en estado **Correcto** hacer clic en siguiente y luego hacemos clic en instalación y esperamos que termine la instalación, luego clic en siguiente y finalmente en cerrar.

NOTA: Si todas las reglas no están en estado **Correcto** no se podrá seguir con la instalación (para ver las reglas clic en el botón mostrar detalles).

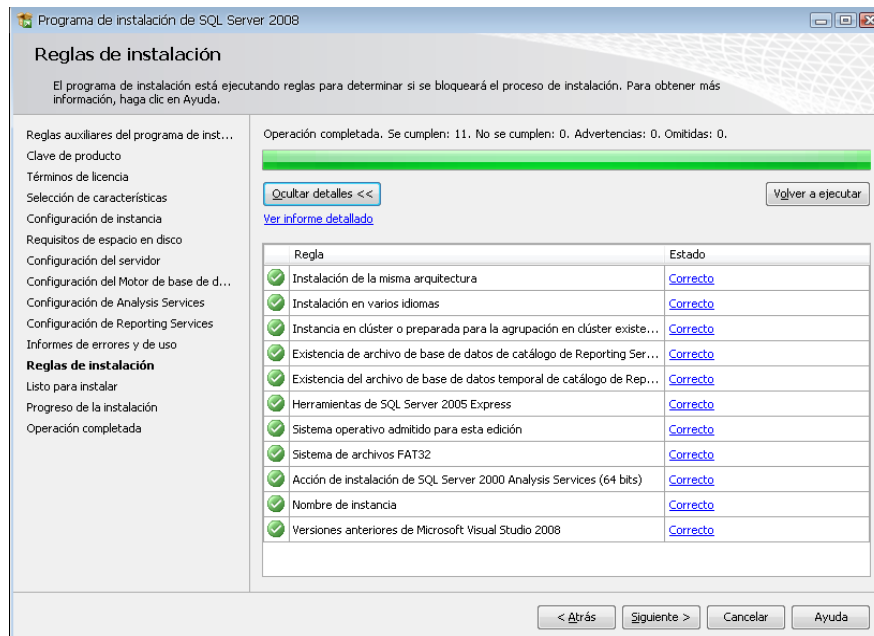
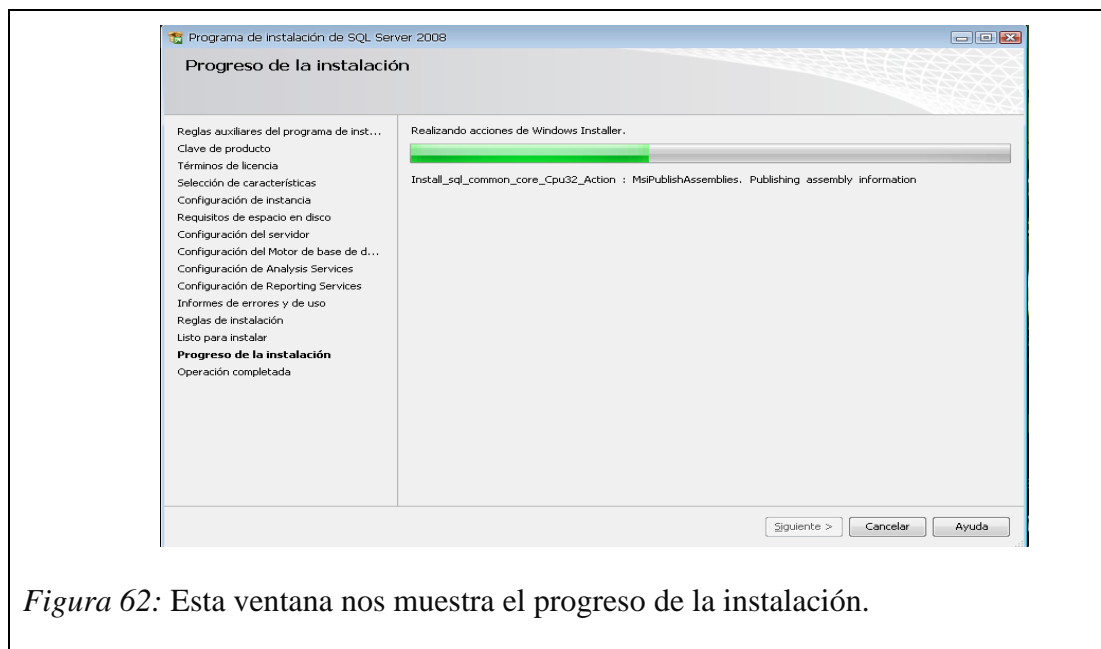
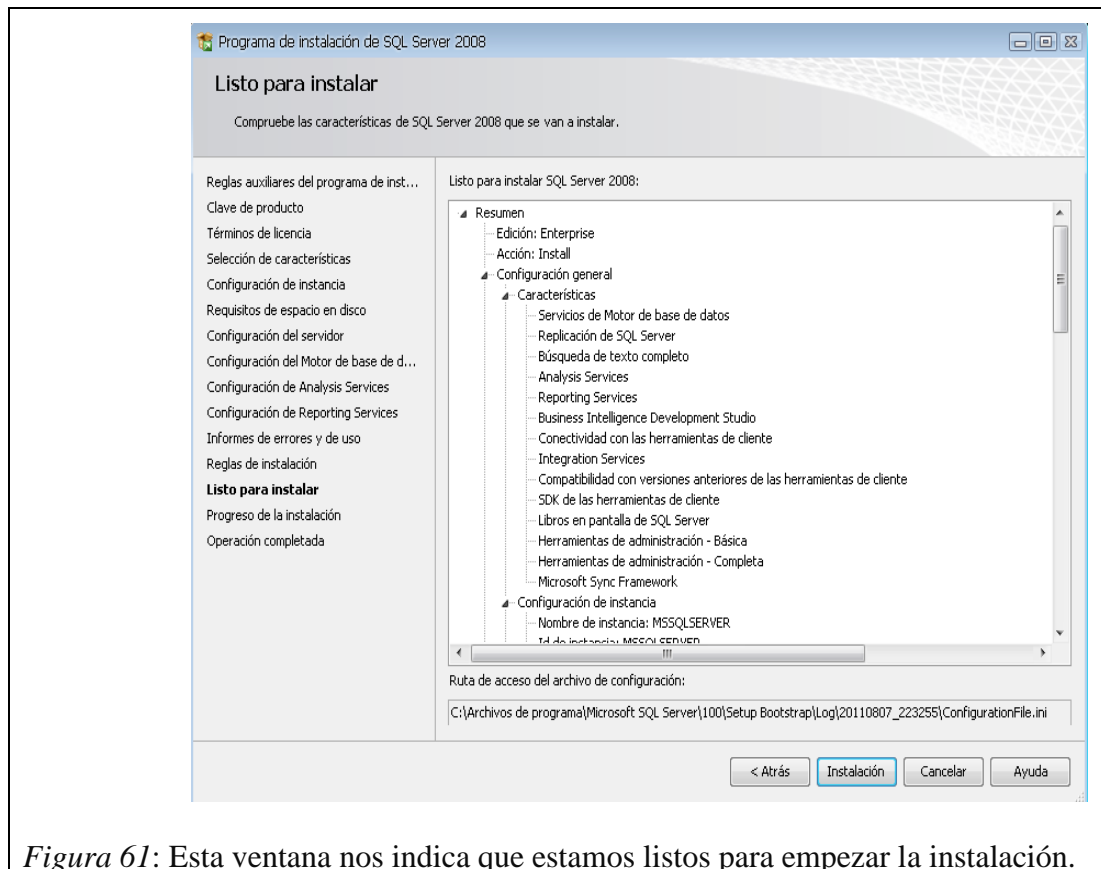


Figura 60: En esta ventana nos mostrara las reglas necesarias para la instalación.



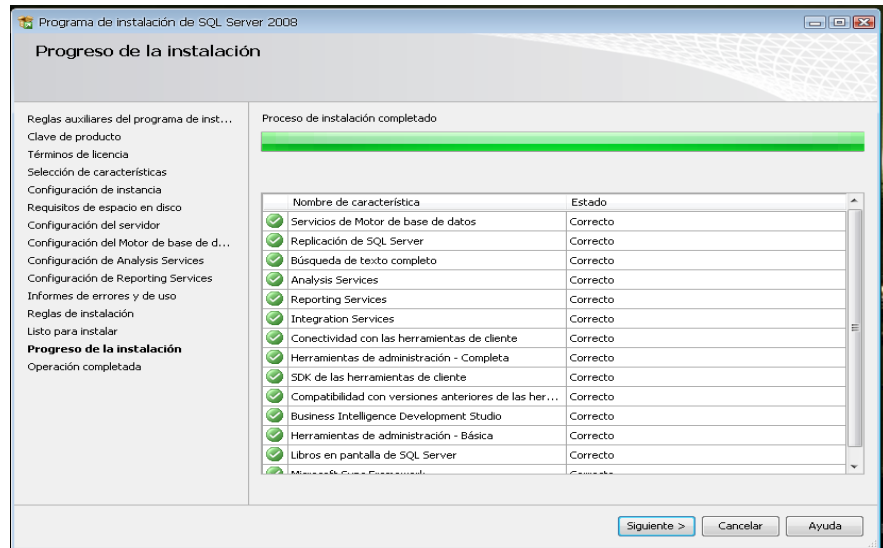


Figura 63: Progreso de la instalación

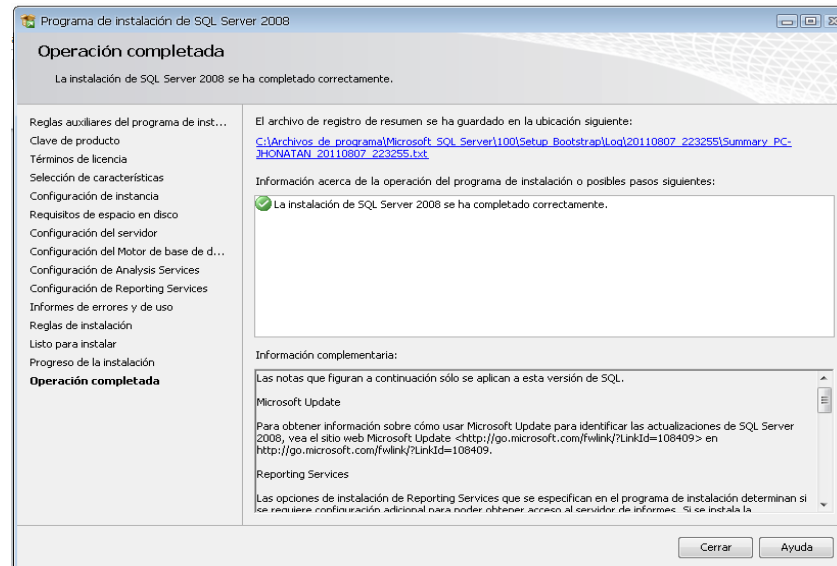


Figura 64: Esta ventana nos indica que la instalación se realizó correctamente.

13 En la opción Autenticación seleccionamos “Autenticación de SQL Server”, en inicio de sesión “sa” y escribimos la contraseña (contraseña que escribimos



cuando configuramos el Motor de Base de Datos – **PASO N° 10**).

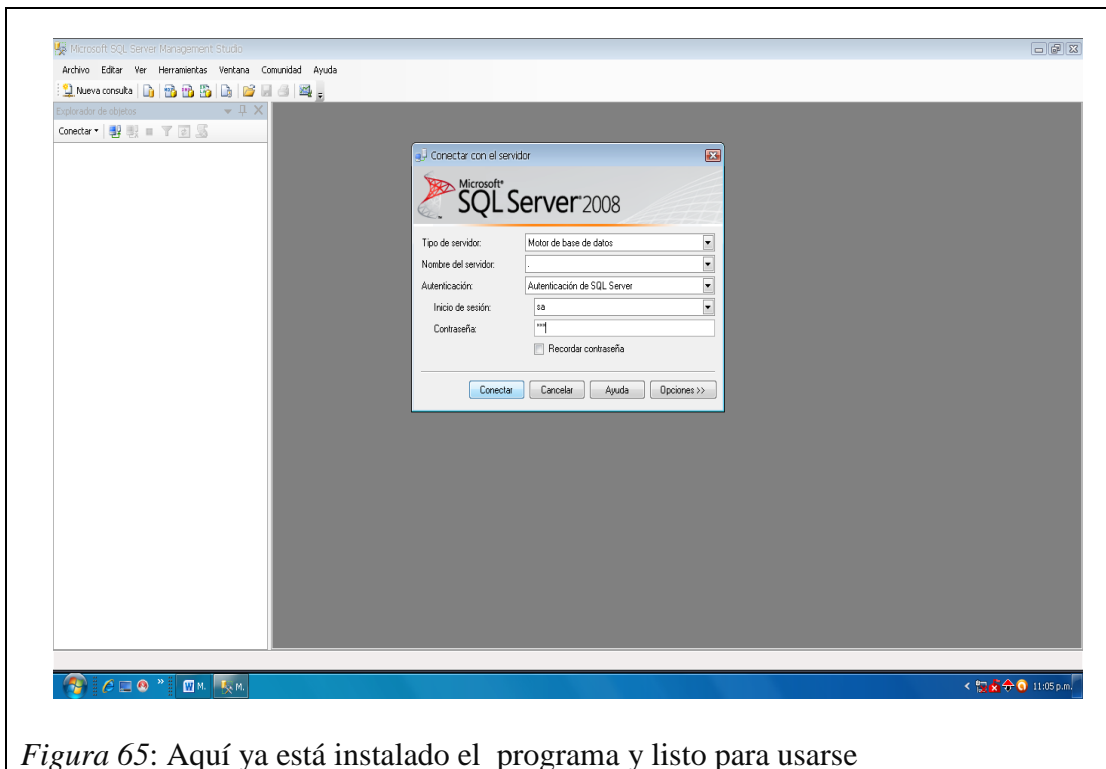


Figura 65: Aquí ya está instalado el programa y listo para usarse

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



SCRIPT DE LA BASE DE DATOS

```
USE [DB_BARRIO]
GO
/***** Object: Table [dbo].[TBL_TIPODOCUMENTO]   Script Date:
04/10/2014 00:19:09 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_TIPODOCUMENTO](
    [id_tipodoc] [int] NOT NULL,
    [des_tipodoc] [varchar](20) NULL,
    [est_tipodoc] [char](1) NULL,
    PRIMARY KEY CLUSTERED
(
    [id_tipodoc] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_TIPO]   Script Date: 04/10/2014 00:19:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_TIPO](
    [id_tipo] [int] NOT NULL,
    [des_tipo] [varchar](50) NULL,
    [est_tipo] [char](1) NULL,
    [cos_tipo] [decimal](5, 2) NULL,
    PRIMARY KEY CLUSTERED
(
    [id_tipo] ASC
```



```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO  
SET ANSI_PADDING OFF  
GO  
/***** Object: Table [dbo].[TBL_SOCIO]   Script Date: 04/10/2014  
00:19:09 *****/  
SET ANSI_NULLS ON  
GO  
SET QUOTED_IDENTIFIER ON  
GO  
SET ANSI_PADDING ON  
GO  
CREATE TABLE [dbo].[TBL_SOCIO](  
    [id_socio] [int] NOT NULL,  
    [ced_socio] [varchar](13) NOT NULL,  
    [ape_socio] [varchar](30) NOT NULL,  
    [nom_socio] [varchar](30) NOT NULL,  
    [dir_socio] [varchar](100) NOT NULL,  
    [tel_socio] [char](9) NULL,  
    [cel_socio] [char](10) NULL,  
    [est_socio] [char](1) NOT NULL,  
    [gen_socio] [char](1) NULL,  
    [fen_socio] [datetime] NULL,  
PRIMARY KEY CLUSTERED  
(  
    [id_socio] ASC  
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO  
SET ANSI_PADDING OFF  
GO  
/***** Object: Table [dbo].[TBL_ROL]   Script Date: 04/10/2014 00:19:09  
*****/  
SET ANSI_NULLS ON  
GO  
SET QUOTED_IDENTIFIER ON  
GO  
SET ANSI_PADDING ON  
GO
```



```
CREATE TABLE [dbo].[TBL_ROL](
    [id_rol] [int] NOT NULL,
    [des_rol] [varchar](50) NOT NULL,
    [est_rol] [char](1) NOT NULL,
    PRIMARY KEY CLUSTERED
(
    [id_rol] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_VITACORA]    Script Date: 04/10/2014
00:19:09 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_VITACORA](
    [id_vitacora] [int] NOT NULL,
    [fec_vitacora] [datetime] NULL,
    [id_usuario] [int] NULL,
    [accion_vit] [varchar](50) NULL,
    [objeto_vit] [varchar](50) NULL,
    [ip_vit] [varchar](20) NULL,
    [estacion_vit] [varchar](10) NULL,
    [est_vit] [char](1) NULL,
    PRIMARY KEY CLUSTERED
(
    [id_vitacora] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_USUARIO]    Script Date: 04/10/2014
00:19:09 *****/
SET ANSI_NULLS ON
```




```
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_USUARIO](
    [id_usuario] [int] NOT NULL,
    [ced_usuario] [char](10) NOT NULL,
    [ape_usuario] [varchar](40) NOT NULL,
    [nom_usuario] [varchar](40) NOT NULL,
    [dir_usuario] [varchar](100) NULL,
    [tel_usuario] [char](9) NULL,
    [cel_usuario] [char](10) NULL,
    [log_usuario] [varchar](25) NULL,
    [pas_usuario] [varchar](30) NULL,
    [est_usuario] [char](1) NOT NULL,
    [id_rol] [int] NOT NULL,
    PRIMARY KEY CLUSTERED
(
    [id_usuario] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_FACTURA]    Script Date: 04/10/2014
00:19:09 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_FACTURA](
    [id_factura] [int] NOT NULL,
    [fec_factura] [datetime] NOT NULL,
    [num_factura] [int] NOT NULL,
    [tot_factura] [decimal](6, 2) NULL,
    [est_factura] [char](1) NOT NULL,
    [id_socio] [int] NOT NULL,
    PRIMARY KEY CLUSTERED
(
```



```
[id_factura] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_LOTE]    Script Date: 04/10/2014 00:19:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_LOTE](
    [id_lote] [int] NOT NULL,
    [num_lote] [int] NULL,
    [mtr_lote] [numeric](5, 2) NULL,
    [est_lote] [char](1) NULL,
    [mnz_lote] [varchar](2) NULL,
    [id_socio] [int] NOT NULL,
    PRIMARY KEY CLUSTERED
(
    [id_lote] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_DFACTURA]    Script Date: 04/10/2014
00:19:09 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_DFACTURA](
    [id_dfactura] [int] NOT NULL,
    [can_dfactura] [int] NULL,
```



```
[puni_dfactura] [decimal](6, 2) NULL,  
[vto_dfactura] [decimal](6, 2) NULL,  
[est_dfactura] [char](1) NOT NULL,  
[id_factura] [int] NOT NULL,  
[id_tipo] [int] NOT NULL,  
[des_dfactura] [varchar](40) NULL,  
CONSTRAINT [PK__TBL_DFAC__ACE922A507020F21] PRIMARY KEY  
CLUSTERED  
(  
    [id_dfactura] ASC  
) WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
    ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO  
SET ANSI_PADDING OFF  
GO  
/***** Object: Table [dbo].[TBL_HISTORIAL]    Script Date: 04/10/2014  
00:19:09 *****/  
SET ANSI_NULLS ON  
GO  
SET QUOTED_IDENTIFIER ON  
GO  
SET ANSI_PADDING ON  
GO  
CREATE TABLE [dbo].[TBL_HISTORIAL](  
    [id_historial] [int] NOT NULL,  
    [fec_historial] [datetime] NULL,  
    [num_historial] [int] NULL,  
    [est_historial] [char](1) NULL,  
    [obs_historial] [varchar](100) NULL,  
    [arc_historial] [varbinary](100) NULL,  
    [id_usuario] [int] NOT NULL,  
    PRIMARY KEY CLUSTERED  
(  
        [id_historial] ASC  
) WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
    ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO  
SET ANSI_PADDING OFF  
GO
```



```

/***** Object: Table [dbo].[TBL_DOCUMENTOSRECIBIDOS]  Script
Date: 04/10/2014 00:19:09 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_DOCUMENTOSRECIBIDOS](
    [id_recibidos] [int] NOT NULL,
    [des_recibidos] [varchar](100) NOT NULL,
    [num_recibidos] [varchar](25) NOT NULL,
    [est_recibidos] [char](1) NULL,
    [id_usuario] [int] NOT NULL,
    [fec_recep] [datetime] NULL,
    [destinatario_rec] [varchar](20) NULL,
    [fec_entre] [datetime] NULL,
    [arc_recibidos] [varbinary](50) NULL,
    PRIMARY KEY CLUSTERED
(
    [id_recibidos] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[TBL_DOCUMENTOSENTREGADOS]
ScriptDate: 04/10/2014 00:19:09 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS](
    [id_entregados] [int] NOT NULL,
    [des_entregados] [varchar](100) NULL,
    [fec_entregados] [datetime] NULL,
    [est_entregados] [char](1) NULL,
    [id_usuario] [int] NOT NULL,
    [id_recibidos] [int] NOT NULL,
    [id_tipodoc] [int] NULL,

```



```
CONSTRAINT [PK__TBL_DOCU__EEB7EA221A14E395] PRIMARY KEY
CLUSTERED
(
    [id_entregados] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
SET ANSI_PADDING OFF
GO
/***** Object: ForeignKey [FK__TBL_DFACT__id_fa__2D27B809]
Script Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_DFACTURA] WITH CHECK ADD
CONSTRAINT [FK__TBL_DFACT__id_fa__2D27B809] FOREIGN
KEY([id_factura])
REFERENCES [dbo].[TBL_FACTURA] ([id_factura])
GO
ALTER TABLE [dbo].[TBL_DFACTURA] CHECK CONSTRAINT
[FK__TBL_DFACT__id_fa__2D27B809]
GO
/***** Object: ForeignKey [FK__TBL_DFACT__id_ti__32E0915F]   Script
Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_DFACTURA] WITH CHECK ADD
CONSTRAINT [FK__TBL_DFACT__id_ti__32E0915F] FOREIGN
KEY([id_tipo])
REFERENCES [dbo].[TBL_TIPO] ([id_tipo])
GO
ALTER TABLE [dbo].[TBL_DFACTURA] CHECK CONSTRAINT
[FK__TBL_DFACT__id_ti__32E0915F]
GO
/***** Object: ForeignKey [FK__TBL_DOCUM__id_re__31EC6D26]
Script Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS] WITH
CHECK ADD CONSTRAINT [FK__TBL_DOCUM__id_re__31EC6D26]
FOREIGN KEY([id_recibidos])
REFERENCES [dbo].[TBL_DOCUMENTOSRECIBIDOS] ([id_recibidos])
GO
ALTER TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS] CHECK
CONSTRAINT [FK__TBL_DOCUM__id_re__31EC6D26]
GO
/***** Object: ForeignKey [FK__TBL_DOCUM__id_ti__33D4B598]
Script Date: 04/10/2014 00:19:09 *****/
```



```
ALTER TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS] WITH
CHECK ADD CONSTRAINT [FK__TBL_DOCUM__id_ti__33D4B598]
FOREIGN KEY([id_tipodoc])
REFERENCES [dbo].[TBL_TIPODOCUMENTO] ([id_tipodoc])
GO
ALTER TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS] CHECK
CONSTRAINT [FK__TBL_DOCUM__id_ti__33D4B598]
GO
/***** Object: ForeignKey [FK__TBL_DOCUM__id_us__2F10007B]
Script Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS] WITH
CHECK ADD CONSTRAINT [FK__TBL_DOCUM__id_us__2F10007B]
FOREIGN KEY([id_usuario])
REFERENCES [dbo].[TBL_USUARIO] ([id_usuario])
GO
ALTER TABLE [dbo].[TBL_DOCUMENTOSENTREGADOS] CHECK
CONSTRAINT [FK__TBL_DOCUM__id_us__2F10007B]
GO
/***** Object: ForeignKey [FK__TBL_DOCUM__id_us__2E1BDC42]
Script Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_DOCUMENTOSRECIBIDOS] WITH CHECK
ADD FOREIGN KEY([id_usuario])
REFERENCES [dbo].[TBL_USUARIO] ([id_usuario])
GO
/***** Object: ForeignKey [FK__TBL_FACTU__id_so__2C3393D0]
Script Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_FACTURA] WITH CHECK ADD FOREIGN
KEY([id_socio])
REFERENCES [dbo].[TBL_SOCIO] ([id_socio])
GO
/***** Object: ForeignKey [FK__TBL_HISTO__id_us__300424B4] Script
Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_HISTORIAL] WITH CHECK ADD FOREIGN
KEY([id_usuario])
REFERENCES [dbo].[TBL_USUARIO] ([id_usuario])
GO
/***** Object: ForeignKey [FK__TBL_LOTE__id_soc__2B3F6F97] Script
Date: 04/10/2014 00:19:09 *****/
ALTER TABLE [dbo].[TBL_LOTE] WITH CHECK ADD FOREIGN
KEY([id_socio])
REFERENCES [dbo].[TBL_SOCIO] ([id_socio])
GO
/***** Object: ForeignKey [FK__TBL_USUAR__id_ro__30F848ED]
Script Date: 04/10/2014 00:19:09 *****/
```



```
ALTER TABLE [dbo].[TBL_USUARIO] WITH CHECK ADD FOREIGN  
KEY([id_rol])  
REFERENCES [dbo].[TBL_ROL] ([id_rol])  
GO
```

Diccionario de Datos

Un diccionario de datos es un conjunto de metadatos que contiene las características lógicas de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

Estos diccionarios se desarrollan durante el análisis de flujo de datos y ayuda a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto.

Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.

En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema. Los elementos más importantes son flujos de datos, almacenes de datos y procesos. El diccionario de datos guarda los detalles y descripción de todos estos elementos.



Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_DFACTURA	1	id_dfactura	Int	4	N	Y	N
dbo	TBL_DFACTURA	2	can_dfactura	Int	4	Y	N	N
dbo	TBL_DFACTURA	3	puni_dfactura	Decimal (6,2)	5	Y	N	N
dbo	TBL_DFACTURA	4	vto_dfactura	Decimal (6,2)	5	Y	N	N
dbo	TBL_DFACTURA	5	est_dfactura	Char (1)	1	N	N	N
dbo	TBL_DFACTURA	6	id_factura	Int	4	N	N	Y
dbo	TBL_DFACTURA	7	id_tipo	Int	4	N	N	Y
dbo	TBL_DFACTURA	8	des_dfactura	VarChar (40)	40	Y	N	N

Figura 66: Diccionario de datos de la tabla TBL_DFACTURA

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_DOCUMENTOS ENTREGADOS	1	id_entregados	Int	4	N	Y	N
dbo	TBL_DOCUMENTOS ENTREGADOS	2	des_entregados	VarChar (100)	100	Y	N	N
dbo	TBL_DOCUMENTOS ENTREGADOS	3	fec_entregados	DateTime	8	Y	N	N
dbo	TBL_DOCUMENTOS ENTREGADOS	4	est_entregados	Char (1)	1	Y	N	N
dbo	TBL_DOCUMENTOS ENTREGADOS	5	id_usuario	Int	4	N	N	Y
dbo	TBL_DOCUMENTOS ENTREGADOS	6	id_recibidos	Int	4	N	N	Y
dbo	TBL_DOCUMENTOS ENTREGADOS	8	id_tipodoc	Int	4	Y	N	Y

Figura 67: Diccionario de datos de la tabla TBL_DOCUMENTOSENTREGADOS



Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_DOCUMENTOS RECIBIDOS	1	id_recibidos	Int	4	N	Y	N
dbo	TBL_DOCUMENTOS RECIBIDOS	2	des_recibidos	VarChar (100)	100	N	N	N
dbo	TBL_DOCUMENTOS RECIBIDOS	3	num_recibidos	VarChar (25)	25	N	N	N
dbo	TBL_DOCUMENTOS RECIBIDOS	4	est_recibidos	Char (1)	1	Y	N	N
dbo	TBL_DOCUMENTOS RECIBIDOS	5	id_usuario	Int	4	N	N	Y
dbo	TBL_DOCUMENTOS RECIBIDOS	6	fec_recep	DateTime	8	Y	N	N
dbo	TBL_DOCUMENTOS RECIBIDOS	7	destinatario_rec	VarChar (20)	20	Y	N	N
dbo	TBL_DOCUMENTOS RECIBIDOS	8	fec_entre	DateTime	8	Y	N	N
dbo	TBL_DOCUMENTOS RECIBIDOS	9	arc_recibidos	VarBinary	50	Y	N	N

Figura 68:Diccionario de datos de la tabla TBL_DOCUMENTOSRECIBIDOS

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_FACTURA	1	id_factura	Int	4	N	Y	N
dbo	TBL_FACTURA	2	fec_factura	DateTime	8	N	N	N
dbo	TBL_FACTURA	3	num_factura	Int	4	N	N	N
dbo	TBL_FACTURA	6	tot_factura	Decimal (6,2)	5	Y	N	N
dbo	TBL_FACTURA	8	est_factura	Char (1)	1	N	N	N
dbo	TBL_FACTURA	9	id_socio	Int	4	N	N	Y

Figura 69: Diccionario de datos de la tabla TBL_FACTURA



Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_HISTORIAL	1	id_historial	Int	4	N	Y	N
dbo	TBL_HISTORIAL	2	fec_historial	DateTime	8	Y	N	N
dbo	TBL_HISTORIAL	3	num_historial	Int	4	Y	N	N
dbo	TBL_HISTORIAL	4	est_historial	Char (1)	1	Y	N	N
dbo	TBL_HISTORIAL	5	obs_historial	VarChar (100)	100	Y	N	N
dbo	TBL_HISTORIAL	6	arc_historial	VarBinary	100	Y	N	N
dbo	TBL_HISTORIAL	7	id_usuario	Int	4	N	N	Y

Figura 70: Diccionario de datos de la tabla TBL_HISTORIAL

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_LOTE	1	id_lote	Int	4	N	Y	N
dbo	TBL_LOTE	2	num_lote	Int	4	Y	N	N
dbo	TBL_LOTE	3	mtr_lote	Numeric	5	Y	N	N
dbo	TBL_LOTE	4	est_lote	Char (1)	1	Y	N	N
dbo	TBL_LOTE	5	mnz_lote	VarChar (2)	2	Y	N	N
dbo	TBL_LOTE	6	id_socio	Int	4	N	N	Y

Figura 71: Diccionario de datos de la tabla TBL_LOTE

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_ROL	1	id_rol	Int	4	N	Y	N
dbo	TBL_ROL	2	des_rol	VarChar (50)	50	N	N	N
dbo	TBL_ROL	3	est_rol	Char (1)	1	N	N	N

Figura 72: Diccionario de datos de la tabla TBL_ROL

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_SOCIO	1	id_socio	Int	4	N	Y	N
dbo	TBL_SOCIO	2	ced_socio	VarChar (13)	13	N	N	N
dbo	TBL_SOCIO	3	ape_socio	VarChar (30)	30	N	N	N
dbo	TBL_SOCIO	4	nom_socio	VarChar (30)	30	N	N	N
dbo	TBL_SOCIO	5	dir_socio	VarChar (100)	100	N	N	N
dbo	TBL_SOCIO	6	tel_socio	Char (9)	9	Y	N	N
dbo	TBL_SOCIO	7	cel_socio	Char (10)	10	Y	N	N
dbo	TBL_SOCIO	8	est_socio	Char (1)	1	N	N	N
dbo	TBL_SOCIO	9	gen_socio	Char (1)	1	Y	N	N
dbo	TBL_SOCIO	10	fen_socio	DateTime	8	Y	N	N

Figura 73: Diccionario de datos de la tabla TBL_SOCIO.

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_TIPO	1	id_tipo	Int	4	N	Y	N
dbo	TBL_TIPO	2	des_tipo	VarChar (50)	50	Y	N	N
dbo	TBL_TIPO	3	est_tipo	Char (1)	1	Y	N	N
dbo	TBL_TIPO	4	cos_tipo	Decimal (5,2)	5	Y	N	N

Figura 74: Diccionario de datos de la tabla TBL_TIPO

Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_TIPODOCUMENTO	1	id_tipodoc	Int	4	N	Y	N
dbo	TBL_TIPODOCUMENTO	2	des_tipodoc	VarChar (20)	20	Y	N	N
dbo	TBL_TIPODOCUMENTO	3	est_tipodoc	Char (1)	1	Y	N	N

Figura 75: Diccionario de datos de la tabla TBL_TIPODOCUMENTO



Schema	Table	Number	Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey
dbo	TBL_USUARIO	1	id_usuario	Int	4	N	Y	N
dbo	TBL_USUARIO	2	ced_usuario	Char (10)	10	N	N	N
dbo	TBL_USUARIO	3	ape_usuario	VarChar (40)	40	N	N	N
dbo	TBL_USUARIO	4	nom_usuario	VarChar (40)	40	N	N	N
dbo	TBL_USUARIO	5	dir_usuario	VarChar (100)	100	Y	N	N
dbo	TBL_USUARIO	6	tel_usuario	Char (9)	9	Y	N	N
dbo	TBL_USUARIO	7	cel_usuario	Char (10)	10	Y	N	N
dbo	TBL_USUARIO	8	log_usuario	VarChar (25)	25	Y	N	N
dbo	TBL_USUARIO	9	pas_usuario	VarChar (30)	30	Y	N	N
	TBL_USUARIO	10	est_usuario	Char (1)	1	N	N	N
	TBL_USUARIO	11	id_rol	Int	4	N	N	Y

Figura 76: Diccionario de datos de la tabla TBL_USUARIO

MANUAL TÉCNICO

LÓGICA DE NEGOCIOS

Logica_Usuario

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Acceso_Datos;
using System.Data.Linq;

namespace Logica_Negocio
{
    public class Logica_Usuario
    {
        public static Dbml_barrioDataContext dc = new Dbml_barrioDataContext();
        public static List<TBL_USUARIO> ObtenerUsuariosLista()
        {
            var lista = dc.TBL_USUARIO.Where(u => u.est_usuario == 'A');
            return lista.ToList();
        }
    }
}
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
public static List<TBL_USUARIO> ObtenerUsuariosIDLista(int Id)
{
    var lista = dc.TBL_USUARIO.Where(u => u.est_usuario == 'A' &
u.id_usuario.Equals(Id));
    return lista.ToList();
}
public static List<TBL_USUARIO> ObtenerUsuariosXCedula(string cedula)
{
    var lista = dc.TBL_USUARIO.Where(u => u.est_usuario == 'A' &
u.ced_usuario.Equals(cedula));
    return lista.ToList();
}
public static List<TBL_USUARIO> ObtenerUsuariosXApellido(string
apellido)
{
    var lista = dc.TBL_USUARIO.Where(u => u.est_usuario == 'A' &
u.ape_usuario.Equals(apellido));
    return lista.ToList();
}
public static TBL_USUARIO ObtenerUsuariosXCedulaInfo(string cedula)
{
    var lista = dc.TBL_USUARIO.FirstOrDefault(u => u.est_usuario == 'A' &
u.ced_usuario.Equals(cedula));           //FirstOrDefault trae el primer
dato de la celda
    return lista;
}
public static TBL_USUARIO ObtenerUsuariosXIdInfo(int Id)
{
    TBL_USUARIO usu = dc.TBL_USUARIO.FirstOrDefault(u =>
u.est_usuario == 'A' & u.id_usuario.Equals(Id));           //FirstOrDefault
trae el primer dato de la celda
    return usu;
}
public static List<TBL_USUARIO> ObtenerUsuariosXTipoUsuario(int
id_TipoUsuario)
{
    var lista = dc.TBL_USUARIO.Where(u => u.est_usuario == 'A' &
u.id_rol.Equals(id_TipoUsuario));
    return lista.ToList();
}
public static int idcodigo()
{
    try
    {
```



```
        return dc.TBL_USUARIO.Max(u => u.id_usuario + 1);
    }
    catch (Exception)
    {

        return 1;
    }

}

public static void Save(TBL_USUARIO UsuarioInfo)
{
    try
    {
        Validaciones(UsuarioInfo);
        dc.ExecuteCommand("Insert into TBL_USUARIO(id_usuario,
ced_usuario, ape_usuario, nom_usuario, dir_usuario, tel_usuario, cel_usuario,
log_usuario, pas_usuario, est_usuario, id_rol)" + "
values({0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10})",//ExecuteCommand
permite ejecutar codigo SQL transformando a objetos x ejemplo insert into,
update,delete
        new object[]
        {
            UsuarioInfo.id_usuario= idcodigo(),
            UsuarioInfo.ced_usuario,
            UsuarioInfo.ape_usuario,
            UsuarioInfo.nom_usuario,
            UsuarioInfo.dir_usuario,
            UsuarioInfo.tel_usuario,
            UsuarioInfo.cel_usuario,
            UsuarioInfo.log_usuario,
            UsuarioInfo.pas_usuario,
            UsuarioInfo.est_usuario='A',

            UsuarioInfo.id_rol

        });
    }
    catch (Exception ex)
    {

        throw new ArgumentException("Datos guardados<br>" + ex.Message);
    }
}
```



```
public static void Modify(TBL_USUARIO UsuarioInfo)
{
    dc.ExecuteCommand("Update TBL_USUARIO set ced_usuario={0},
ape_usuario={1}, nom_usuario={2}, dir_usuario={3}, tel_usuario={4},
cel_usuario={5}, log_usuario={6}, pas_usuario={7}, est_usuario={8}, id_rol={9}
where id_usuario={10}", //ExecuteCommand permite ejecutar código SQL
transformando a objetos x ejemplo insert into, update, delete
    new object[]
    {
        UsuarioInfo.ced_usuario,
        UsuarioInfo.ape_usuario,
        UsuarioInfo.nom_usuario,
        UsuarioInfo.dir_usuario,
        UsuarioInfo.tel_usuario,
        UsuarioInfo.cel_usuario,
        UsuarioInfo.log_usuario,
        UsuarioInfo.pas_usuario,
        UsuarioInfo.est_usuario='A',
        UsuarioInfo.id_rol,
        UsuarioInfo.id_usuario
    });
    dc.Refresh(RefreshMode.OverwriteCurrentValues, UsuarioInfo);
}
public static void Delete(TBL_USUARIO UsuarioInfo)
{
    try
    {
        dc.ExecuteCommand("Update TBL_USUARIO set est_usuario={0}
where id_usuario={1}",
            new object[]
            {
                UsuarioInfo.est_usuario= 'E',
                UsuarioInfo.id_usuario
            });
        dc.Refresh(RefreshMode.OverwriteCurrentValues, UsuarioInfo);
    }
    catch (Exception ex)
    {
        throw new ArgumentException("Datos no eliminados<br>" +
ex.Message);
    }
}
private static void Validaciones(TBL_USUARIO UsuarioInfo)
```



```
{
    string sms="";
    if (UsuarioInfo.ape_usuario=="")
    {
        sms += "Ingresar apellido paterno</br>";
    }
    if (UsuarioInfo.ced_usuario == "")
    {
        sms += "Ingresar cedula</br>";
    }
    if (UsuarioInfo.nom_usuario == "")
    {
        sms += "Ingresar nombre</br>";
    }
    if (sms != "")
    {
        throw new ArgumentException(sms);
    }
}

public static bool Existe (string nom, string pass)
{
    var Existe = dc.TBL_USUARIO.Any(E => E.est_usuario == 'A' &
E.log_usuario.Equals(nom) & E.pas_usuario.Equals(pass));
    return Existe;
}

public static TBL_USUARIO UsuarioActualyanoExiste(string nom, string
pass)
{
    TBL_USUARIO UsuarioActualExiste =
dc.TBL_USUARIO.FirstOrDefault(u => u.est_usuario == 'A' &
u.log_usuario.Equals(nom) &
u.pas_usuario.Equals(pass));

    return UsuarioActualExiste;
}
}
}
```

Logica_rol

using System;

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Acceso_Datos;
using System.Data.Linq;

namespace Logica_Negocio
{
    public class Logica_Rol
    {
        public static Dbml_barrioDataContext dc = new Dbml_barrioDataContext();
        public static List<TBL_ROL> ObtenerTipoUsuarioLista()
        {
            var lista = dc.TBL_ROL.Where(u => u.est_rol == 'A');
            return lista.ToList();
        }
        public static List<TBL_ROL> ObtenerTipoUsuarioIDLista(int Id)
        {
            var lista = dc.TBL_ROL.Where(u => u.est_rol == 'A' & u.id_rol.Equals(Id));
            return lista.ToList();
        }
        public static List<TBL_ROL> ObtenerUsuariosXNombre(string nombre)
        {
            var lista = dc.TBL_ROL.Where(u => u.est_rol == 'A' &
u.des_rol.Equals(nombre));
            return lista.ToList();
        }
        public static TBL_ROL ObtenerTipoUsuarioXIdInfo(int Id)
        {
            TBL_ROL usu = dc.TBL_ROL.FirstOrDefault(u => u.est_rol == 'A' &
u.id_rol.Equals(Id));
            return usu;
        }

        public static int idcodigo()
        {
            try
            {
                return dc.TBL_ROL.Max(u => u.id_rol + 1);
            }
            catch (Exception)
            {
                return 1;
            }
        }
    }
}
```



```
    }  
  
    }  
    public static void Save(TBL_ROL TipoUsuarioInfo)  
    {  
        try  
        {  
            Validaciones(TipoUsuarioInfo);  
            dc.ExecuteCommand("Insert into TBL_ROL (id_rol, des_rol, est_rol)" + "  
values({0},{1},{2})",  
            new object[]  
            {  
                TipoUsuarioInfo.id_rol= idcodigo(),  
  
                TipoUsuarioInfo.des_rol,  
                TipoUsuarioInfo.est_rol='A'  
  
            });  
  
        }  
        catch (Exception ex)  
        {  
  
            throw new ArgumentException("Datos no guardados<br>" +  
ex.Message);  
        }  
    }  
    public static void Modify(TBL_ROL TipoUsuarioInfo)  
    {  
        dc.ExecuteCommand("Update TBL_ROL set des_rol={0}, est_rol={1}  
where id_rol={2}",  
            new object[]  
            {  
                TipoUsuarioInfo.des_rol,  
                TipoUsuarioInfo.est_rol,  
                TipoUsuarioInfo.id_rol  
            });  
  
    }  
    public static void Delete(TBL_ROL TipoUsuarioInfo)  
    {  
        dc.ExecuteCommand("Update TBL_ROL set est_rol='E' where id_rol={0}",  
            new object[]  
            {
```



```
TipoUsuarioInfo.id_rol

    });
    dc.Refresh(RefreshMode.OverwriteCurrentValues, TipoUsuarioInfo);
}
private static void Validaciones(TBL_ROL UsuarioInfo)
{
    string sms="";
    if (UsuarioInfo.des_rol=="")
    {
        sms += "Ingresar nombre de tipo usuario<br>";
    }

    if (sms != "")
    {
        throw new ArgumentException(sms);
    }
}
}
}
```

Logica_Lote

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Acceso_Datos;
using System.Data.Linq;

namespace Logica_Negocio
{
    public class Logica_Lote
    {
        public static Dbml_barrioDataContext dc = new Dbml_barrioDataContext();
        public static List<TBL_LOTE> ObtenerLoteLista()
        {
            var lista = dc.TBL_LOTE.Where(u => u.est_lote == 'A');
            return lista.ToList();
        }
        public static List<TBL_LOTE> ObtenerLoteIDLista(int Id)
        {

```



```
        var lista = dc.TBL_LOTE.Where(u => u.est_lote == 'A' &
u.id_lote.Equals(Id));
        return lista.ToList();
    }
    public static List<TBL_LOTE> ObtenerLoteXNumero(string num)
    {
        var lista = dc.TBL_LOTE.Where(u => u.est_lote == 'A' &
u.num_lote.Equals(num));
        return lista.ToList();
    }
    public static List<TBL_LOTE> ObtenerLoteXManzana(string mnz)
    {
        var lista = dc.TBL_LOTE.Where(u => u.est_lote == 'A' &
u.mnz_lote.Equals(mnz));
        return lista.ToList();
    }

    public static TBL_LOTE ObtenerLoteXIdInfo(int Id)
    {
        TBL_LOTE usu = dc.TBL_LOTE.FirstOrDefault(u => u.est_lote == 'A' &
u.id_lote.Equals(Id));
        return usu;
    }
    public static List<TBL_LOTE> ObtenerLoteXSocio(string id_socio)
    {
        var lista = dc.TBL_LOTE.Where(u => u.est_lote == 'A' &
u.TBL_SOCIO.ape_socio.Equals(id_socio));
        return lista.ToList();
    }
    public static int idcodigo()
    {
        try
        {
            return dc.TBL_LOTE.Max(u => u.id_lote + 1);
        }
        catch (Exception)
        {
            return 1;
        }
    }

    public static void Save(TBL_LOTE LoteInfo)
    {
```



```
try
{
    Validaciones(LoteInfo);
    dc.ExecuteCommand("Insert into TBL_LOTE(id_lote, num_lote, mtr_lote,
est_lote, mnz_lote, id_socio)" + "
values({0},{1},{2},{3},{4},{5})",//ExecuteCommand permite ejecutar codigo SQL
transformando a objetos x ejemplo insert into, update,delete
    new object[]
    {
        LoteInfo.id_lote= idcodigo(),
        LoteInfo.num_lote,
        LoteInfo.mtr_lote,
        LoteInfo.est_lote,
        LoteInfo.mnz_lote,
        LoteInfo.id_socio
    });
}
catch (Exception ex)
{
    throw new ArgumentException("Datos guardados<br>" + ex.Message);
}
}
public static void Modify(TBL_LOTE LoteInfo)
{
    dc.ExecuteCommand("Update TBL_LOTE set num_lote={0}, mtr_lote={1},
est_lote={2}, mnz_lote={3}, id_socio={4} where id_lote={5}",
    new object[]
    {
        LoteInfo.num_lote,
        LoteInfo.mtr_lote,
        LoteInfo.est_lote,
        LoteInfo.mnz_lote,
        LoteInfo.id_socio,
        LoteInfo.id_lote
    });
    dc.Refresh(RefreshMode.OverwriteCurrentValues, LoteInfo);
}
public static void Delete(TBL_LOTE LoteInfo)
{
    try
    {
```



```
        dc.ExecuteCommand("Update TBL_LOTE set est_lote={0} where  
id_lote={1}",  
        new object[]  
        {  
            LoteInfo.est_lote= 'E',  
            LoteInfo.id_lote  
        });  
        dc.Refresh(RefreshMode.OverwriteCurrentValues, LoteInfo);  
    }  
    catch (Exception ex)  
    {  
  
        throw new ArgumentException("Datos no eliminados</br>" +  
ex.Message);  
    }  
}  
private static void Validaciones(TBL_LOTE LoteInfo)  
{  
    string sms = "";  
    if (LoteInfo.num_lote == 0)  
    {  
        sms += "Ingresar el numero de lote</br>";  
    }  
    if (LoteInfo.mtr_lote == 0)  
    {  
        sms += "Ingresar los metros del lote</br>";  
    }  
    if (LoteInfo.mnz_lote == "")  
    {  
        sms += "Ingresar la manzana del lote</br>";  
    }  
    if (sms != "")  
    {  
        throw new ArgumentException(sms);  
    }  
}  
}
```

Logica_TipoDocumento

```
using System;  
using System.Collections.Generic;
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
using System.Linq;
using System.Text;
using Acceso_Datos;
using System.Data.Linq;

namespace Logica_Negocio
{
    public class Logica_TipoDocumento
    {
        public static Dbml_barrioDataContext dc = new Dbml_barrioDataContext();
        public static List<TBL_TIPODOCUMENTO> ObtenerTipoDocLista()
        {
            var lista = dc.TBL_TIPODOCUMENTO.Where(u => u.est_tipodoc == 'A');
            return lista.ToList();
        }
        public static List<TBL_TIPODOCUMENTO> ObtenerTipoIDLista(int Id)
        {
            var lista = dc.TBL_TIPODOCUMENTO.Where(u => u.est_tipodoc == 'A' &
u.id_tipodoc.Equals(Id));
            return lista.ToList();
        }
        public static List<TBL_TIPODOCUMENTO>
ObtenerTipoDcoXDescripcion(string des)
        {
            var lista = dc.TBL_TIPODOCUMENTO.Where(u => u.est_tipodoc == 'A' &
u.des_tipodoc.Equals(des));
            return lista.ToList();
        }

        public static TBL_TIPODOCUMENTO ObtenerTipoDocXIdInfo(int Id)
        {
            TBL_TIPODOCUMENTO usu =
dc.TBL_TIPODOCUMENTO.FirstOrDefault(u => u.est_tipodoc == 'A' &
u.id_tipodoc.Equals(Id));
            return usu;
        }

        public static int idcodigo()
        {
            try
            {
                return dc.TBL_TIPODOCUMENTO.Max(u => u.id_tipodoc + 1);
            }
            catch (Exception)
            {
            }
        }
    }
}
```



```
{
    return 1;
}

}
public static void Save(TBL_TIPODOCUMENTO TipoDocInfo)
{
    try
    {
        Validaciones(TipoDocInfo);
        dc.ExecuteCommand("Insert into TBL_TIPODOCUMENTO(id_tipodoc,
des_tipodoc, est_tipodoc)" + " values({0},{1},{2})",
        new object[]
        {
            TipoDocInfo.id_tipodoc= idcodigo(),
            TipoDocInfo.des_tipodoc,
            TipoDocInfo.est_tipodoc,
        });
    }
    catch (Exception ex)
    {
        throw new ArgumentException("Datos guardados<br>" + ex.Message);
    }
}
public static void Modify(TBL_TIPODOCUMENTO TipoDocInfo)
{
    dc.ExecuteCommand("Update TBL_TIPODOCUMENTO set
des_tipodoc={0}, est_tipodoc={1} where id_tipodoc={2}",
        new object[]
        {
            TipoDocInfo.des_tipodoc,
            TipoDocInfo.est_tipodoc,
            TipoDocInfo.id_tipodoc
        });
    dc.Refresh(RefreshMode.OverwriteCurrentValues, TipoDocInfo);
}
public static void Delete(TBL_TIPODOCUMENTO TipoDocInfo)
{
    try
    {
```




```
        dc.ExecuteCommand("Update TBL_TIPODOCUMENTO set  
est_tipodoc={0} where id_tipodoc={1}",  
        new object[]  
        {  
            TipoDocInfo.est_tipodoc= 'E',  
            TipoDocInfo.id_tipodoc  
        });  
        dc.Refresh(RefreshMode.OverwriteCurrentValues, TipoDocInfo);  
    }  
    catch (Exception ex)  
    {  
  
        throw new ArgumentException("Datos no eliminados</br>" +  
ex.Message);  
    }  
}  
private static void Validaciones(TBL_TIPODOCUMENTO TipoDocInfo)  
{  
    string sms = "";  
    if (TipoDocInfo.des_tipodoc == "")  
    {  
        sms += "Ingresar la descripcion</br>";  
    }  
  
    if (sms != "")  
    {  
        throw new ArgumentException(sms);  
    }  
}  
}  
}  
Logica_Socio
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using Acceso_Datos;  
using System.Data.Linq;
```

```
namespace Logica_Negocio  
{  
    public class Logica_Socio
```



```
{
    public static Dbml_barrioDataContext dc = new Dbml_barrioDataContext();
    public static List<TBL_SOCIO> ObtenerSocioLista()
    {
        var lista = dc.TBL_SOCIO.Where(u => u.est_socio == 'A');
        return lista.ToList();
    }
    public static List<TBL_SOCIO> ObtenerSocioIDLista(int Id)
    {
        var lista = dc.TBL_SOCIO.Where(u => u.est_socio == 'A' &
u.id_socio.Equals(Id));
        return lista.ToList();
    }
    public static List<TBL_SOCIO> ObtenerSocioXCedula(string cedula)
    {
        var lista = dc.TBL_SOCIO.Where(u => u.est_socio == 'A' &
u.ced_socio.Equals(cedula));
        return lista.ToList();
    }
    public static List<TBL_SOCIO> ObtenerSocioXApellido(string apellido)
    {
        var lista = dc.TBL_SOCIO.Where(u => u.est_socio == 'A' &
u.ape_socio.Equals(apellido));
        return lista.ToList();
    }
    public static TBL_SOCIO ObtenerSocioXCedulaInfo(string cedula)
    {
        var lista = dc.TBL_SOCIO.FirstOrDefault(u => u.est_socio == 'A' &
u.ced_socio.Equals(cedula));           //FirstOrDefault trae el primer dato
de la celda
        return lista;
    }
    public static TBL_SOCIO ObtenerSocioXIdInfo(int Id)
    {
        TBL_SOCIO usu = dc.TBL_SOCIO.FirstOrDefault(u => u.est_socio == 'A'
& u.id_socio.Equals(Id));           //FirstOrDefault trae el primer dato de
la celda
        return usu;
    }
    public static int idcodigo()
    {
        try
        {
```



```
        return dc.TBL_SOCIO.Max(u => u.id_socio + 1);
    }
    catch (Exception)
    {

        return 1;
    }

}

public static void Save(TBL_SOCIO SocioInfo)
{
    try
    {
        Validaciones(SocioInfo);
        dc.ExecuteCommand("Insert into TBL_SOCIO(id_socio, ced_socio,
ape_socio, nom_socio, dir_socio, tel_socio, cel_socio, est_socio, gen_socio,
fen_socio)" + " values({0},{1},{2},{3},{4},{5},{6},{7},{8},{9})",
        new object[]
        {
            SocioInfo.id_socio= idcodigo(),
            SocioInfo.ced_socio,
            SocioInfo.ape_socio,
            SocioInfo.nom_socio,
            SocioInfo.dir_socio,
            SocioInfo.tel_socio,
            SocioInfo.cel_socio,
            SocioInfo.est_socio,
            SocioInfo.gen_socio,
            SocioInfo.fen_socio
        });
    }
    catch (Exception ex)
    {

        throw new ArgumentException("Datos guardados<br>" + ex.Message);
    }
}

public static void Modify(TBL_SOCIO SocioInfo)
{
    dc.ExecuteCommand("Update TBL_SOCIO set ced_socio={0},
ape_socio={1}, nom_socio={2}, dir_socio={3}, tel_socio={4}, cel_socio={5},
est_socio={6}, gen_socio={7}, fen_socio={8} where id_socio={9}",
        new object[]
```



```
{
    SocioInfo.ced_socio,
    SocioInfo.ape_socio,
    SocioInfo.nom_socio,
    SocioInfo.dir_socio,
    SocioInfo.tel_socio,
    SocioInfo.cel_socio,
    SocioInfo.est_socio,
    SocioInfo.gen_socio,
    SocioInfo.fen_socio,
    SocioInfo.id_socio
});
dc.Refresh(RefreshMode.OverwriteCurrentValues, SocioInfo);
}
public static void Delete(TBL_SOCIO SocioInfo)
{
    try
    {
        dc.ExecuteCommand("Update TBL_SOCIO set est_socio={0} where
id_socio={1}",
            new object[]
            {
                SocioInfo.est_socio= 'E',
                SocioInfo.id_socio
            });
        dc.Refresh(RefreshMode.OverwriteCurrentValues, SocioInfo);
    }
    catch (Exception ex)
    {
        throw new ArgumentException("Datos no eliminados</br>" +
ex.Message);
    }
}
private static void Validaciones(TBL_SOCIO SocioInfo)
{
    string sms = "";
    if (SocioInfo.ape_socio == "")
    {
        sms += "Ingresar apellido paterno</br>";
    }
    if (SocioInfo.ced_socio == "")
    {
        sms += "Ingresar cedula</br>";
    }
}
```



```
    }  
    if (SocioInfo.nom_socio == "")  
    {  
        sms += "Ingresar nombre</br>";  
    }  
    if (SocioInfo.dir_socio == "")  
    {  
        sms += "Ingresar la direccion</br>";  
    }  
    if (SocioInfo.gen_socio == 0)  
    {  
        sms += "Ingresar el genero</br>";  
    }  
  
    if (sms != "")  
    {  
        throw new ArgumentException(sms);  
    }  
    }  
    }  
}
```

Logica_Tipo

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using Acceso_Datos;  
using System.Data.Linq;  
  
namespace Logica_Negocio  
{  
    public class Logica_Tipo  
    {  
        public static Dbml_barrioDataContext dc = new Dbml_barrioDataContext();  
        public static List<TBL_TIPO> ObtenerTipoLista()  
        {  
            var lista = dc.TBL_TIPO.Where(u => u.est_tipo == 'A');  
            return lista.ToList();  
        }  
        public static List<TBL_TIPO> ObtenerTipoIDLista(int Id)  
        {
```



```
        var lista = dc.TBL_TIPO.Where(u => u.est_tipo == 'A' &
u.id_tipo.Equals(Id));
        return lista.ToList();
    }
    public static List<TBL_TIPO> ObtenerTipoXDescripcion(string des)
    {
        var lista = dc.TBL_TIPO.Where(u => u.est_tipo == 'A' &
u.des_tipo.Equals(des));
        return lista.ToList();
    }

    public static TBL_TIPO ObtenerTipoXIdInfo(int Id)
    {
        TBL_TIPO usu = dc.TBL_TIPO.FirstOrDefault(u => u.est_tipo == 'A' &
u.id_tipo.Equals(Id));
        return usu;
    }

    public static int idcodigo()
    {
        try
        {
            return dc.TBL_TIPO.Max(u => u.id_tipo + 1);
        }
        catch (Exception)
        {
            return 1;
        }
    }

    public static void Save(TBL_TIPO TipoInfo)
    {
        try
        {
            Validaciones(TipoInfo);
            dc.ExecuteCommand("Insert into TBL_TIPO(id_tipo, des_tipo, est_tipo)"
+ " values({0},{1},{2})",
            new object[]
            {
                TipoInfo.id_tipo= idcodigo(),
                TipoInfo.des_tipo,
                TipoInfo.est_tipo
            });
        }
    }
```



```
    }
    catch (Exception ex)
    {

        throw new ArgumentException("Datos guardados</br>" + ex.Message);
    }
}
public static void Modify(TBL_TIPO TipoInfo)
{
    dc.ExecuteCommand("Update TBL_TIPO set des_tipo={0}, est_tipo={1}
where id_tipo={2}",
        new object[]
        {
            TipoInfo.des_tipo,
            TipoInfo.est_tipo,
            TipoInfo.id_tipo
        });
    dc.Refresh(RefreshMode.OverwriteCurrentValues, TipoInfo);
}
public static void Delete(TBL_TIPO TipoInfo)
{
    try
    {
        dc.ExecuteCommand("Update TBL_TIPO set est_tipo={0} where
id_tipo={1}",
            new object[]
            {
                TipoInfo.est_tipo= 'E',
                TipoInfo.id_tipo
            });
        dc.Refresh(RefreshMode.OverwriteCurrentValues, TipoInfo);
    }
    catch (Exception ex)
    {

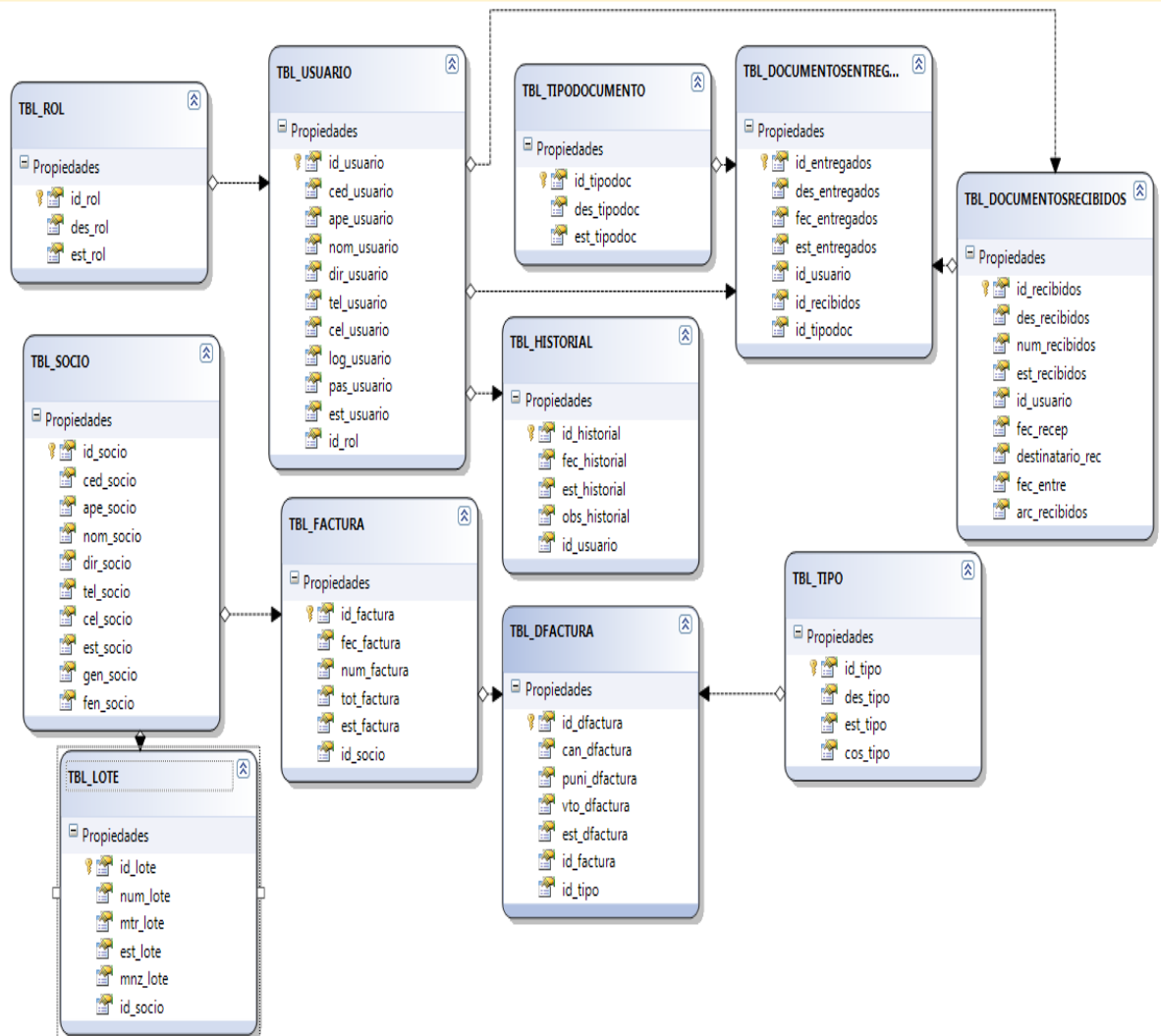
        throw new ArgumentException("Datos no eliminados</br>" +
ex.Message);
    }
}
private static void Validaciones(TBL_TIPO TipoInfo)
{
    string sms = "";
    if (TipoInfo.des_tipo == "")
```



```
{  
    sms += "Ingresar la descripcion</br>";  
}  
  
if (sms != "")  
{  
    throw new ArgumentException(sms);  
}  
}  
}
```

ACCESO DATOS

DBML_BARRIO



CAPA PRESENTACION

Frm_UsuarioNuevo

```

<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_UsuarioNuevo.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Usuario.Frm_UsuarioNuevo" %>
  
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
  <script src="../../Jquerys/jquery-1.7.1.min.js" type="text/javascript"></script>
  <script src="../../Jquerys/jquery.validate.js" type="text/javascript"></script>
  <script type="text/javascript">
```

```
function Validacedula() {
    var cedula = document.getElementById('Txt_cedula').value;
    array = cedula.split("");
    num = array.length;
    if (num == 10) {
        total = 0;
        digito = (array[9] * 1);
        for (i = 0; i < (num - 1); i++) {
            mult = 0;
            if ((i % 2) != 0) {
                total = total + (array[i] * 1);
            }
        }
        else {
            mult = array[i] * 2;
            if (mult > 9)
                total = total + (mult - 9);
            else
                total = total + mult;
        }
    }
    decena = total / 10;
    decena = Math.floor(decena);
    decena = (decena + 1) * 10;
    final = (decena - total);
    if ((final == 10 && digito == 0) || (final == digito)) {
        //alert("La cedula es correcta");
        return true;
    }
    else {
        alert("Cedula Incorrecta");
        document.getElementById('Txt_cedula').value = "";
        document.getElementById('Txt_cedula').focus();
    }
}
```



```
        return false;
    }
}
else {
    alert("La cedula no pue detener menos de 10 digitos y tan poco mas de10
digitos");
    document.getElementById('Txt_cedula').value = "";
    document.getElementById('Txt_cedula').focus();
    return false;
    return false;
}
}
</script>
<script type="text/javascript">
$(document).ready(function () {
    $("#form1").validate({

        rules: {
            'Txt_cedula': { required: true, number: true, minlength: 1 },
            'Txt_direccion': 'required',
            'Txt_telefono': { required: true, number: true, minlength: 1 },
            'Txt_celular': { required: true, number: true, minlength: 1 },
            'Txt_nombre': 'required',

        },
        messages: {
            'Txt_cedula': { required: 'Ingresa solo numeros', number: 'Debe ingresar
datos numericos' },
            'Txt_direccion': 'Ingresa La Direccion',
            'Txt_telefono': { required: 'Ingresa solo numeros', number: 'Debe ingresar
datos numericos' },
            'Txt_celular': { required: 'Ingresa solo numeros', number: 'Debe ingresar
datos numericos' },
            'Txt_nombre': 'Ingresa el nombre',

        },

    //
    });
});
</script>
```



REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL COMITÉ PRO-MEJORAS SANTA ISABEL

APELLIDO:

[illegible]

NOMBRE:

[illegible]

```
<asp:TextBox ID="Txt_password" runat="server"
Width="182px"></asp:TextBox>
<br />
<br />
```

DIRECCION:

[illegible]

TELEFONO:

```
<asp:TextBox ID="Txt_telefono" runat="server"
Width="179px"></asp:TextBox>
    <br />
<br />
<br />
</div>
</div>
</form>
</center>
</body>
</html>
```

Frm UsuarioLista



```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_UsuarioLista.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Usuario.Frm_UsuarioLista" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:ImageButton ID="ImageButton1" runat="server"
                ImageUrl="~/icons/bug_add.png" onclick="ImageButton1_Click" />
            <asp:LinkButton ID="Lnk_nuevo" runat="server"
                onclick="Lnk_nuevo_Click">NUEVO</asp:LinkButton>
            <br />
            LISTA USUARIOS<br />
            &nbsp;<asp:Label ID="lbl_mensaje" runat="server" Font-Bold="true"
                ForeColor="Red"></asp:Label>
            <br />
            CRITERIO DE BUSQUEDA:
            <asp:DropDownList ID="ddl_buscar" runat="server" Width="160 px">
            <asp:ListItem Text="CEDULA" value="c"></asp:ListItem>
            <asp:ListItem Text="APELLIDO" Value="a"></asp:ListItem>
            </asp:DropDownList>
            <asp:TextBox ID="txt_buscar" runat="server" Width="200
                px"></asp:TextBox>
            <asp:ImageButton ID="img_buscar" runat="server"
                ImageUrl="~/icons/application_form_magnify.png"
                onclick="img_buscar_Click"/>
            <asp:DataGrid ID="dgr_usuario" runat="server" CellPadding="4"
                ForeColor="#333333" GridLines="None" AutoGenerateColumns="False"
                onitemcommand="dgr_usuario_ItemCommand">
            <AlternatingItemStyle BackColor="White" />
            <EditItemStyle BackColor="#2461BF" />
            <FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
            <HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White"
            />
            <ItemStyle BackColor="#EFF3FB" />
            <PagerStyle BackColor="#2461BF" ForeColor="White"
                HorizontalAlign="Center" />
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
<SelectedItemStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
<Columns>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEditar" runat="server" CausesValidation="false"
CommandName="Editar"
ImageUrl="~/icons/application_form_edit.png" ToolTip="Editar" Width="16"
/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEliminar" runat="server" CausesValidation="false"
CommandName="Eliminar"
ImageUrl="~/icons/award_star_delete.png" ToolTip="Eliminar" Width="16"
OnClick="return confirm('Seguro que desea eliminar el registro')"/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="Id">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_idusuario" runat="server" Text='<%=Eval("id_usuario")
%>'></asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="CEDULA">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_cedula" runat="server" Text='<%=Eval("ced_usuario")
%>'></asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="APELLIDO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_apellido1" runat="server" Text='<%=Eval("ape_usuario")
%>'></asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="NOMBRE">
<ItemTemplate>
```



```
<span style="float:left;">
  <asp:Label ID="lbl_nombre" runat="server" Text='<#Eval("nom_usuario")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="DIRECCION">
  <ItemTemplate>
    <span style="float:left;">
      <asp:Label ID="lbl_estadocivil" runat="server" Text='<#Eval("dir_usuario")
%>'> </asp:Label>
    </ItemTemplate>
  </asp:TemplateColumn>
  <asp:TemplateColumn HeaderText="TELEFONO">
    <ItemTemplate>
      <span style="float:left;">
        <asp:Label ID="lbl_fecha" runat="server" Text='<#Eval("tel_usuario") %>'>
</asp:Label>
      </ItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn HeaderText="CELULAR">
      <ItemTemplate>
        <span style="float:left;">
          <asp:Label ID="lbl_genero" runat="server" Text='<#Eval("cel_usuario")
%>'> </asp:Label>
        </ItemTemplate>
      </asp:TemplateColumn>
      <asp:TemplateColumn HeaderText="USUARIO">
        <ItemTemplate>
          <span style="float:left;">
            <asp:Label ID="lbl_estado" runat="server" Text='<#Eval("log_usuario")
%>'> </asp:Label>
          </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn HeaderText="PASSWORD">
          <ItemTemplate>
            <span style="float:left;">
              <asp:Label ID="lbl_usuario" runat="server" Text='<#Eval("pas_usuario")
%>'> </asp:Label>
            </ItemTemplate>
          </asp:TemplateColumn>
          <asp:TemplateColumn HeaderText="ESTADO">
            <ItemTemplate>
              <span style="float:left;">
```




```
<asp:Label ID="lbl_password" runat="server" Text='<%#Eval("est_usuario")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="TIPO USUARIO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbltipousuario" runat="server"
Text='<%#Eval("TBL_ROL.des_rol") %>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
</Columns>
</asp:DataGrid>

</div>
</form>
</body>
</html>
```

Frm_RolNuevo

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_RolNuevo.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Usuario.Frm_RolNuevo" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
<asp:Label ID="Lbl_mensaje" runat="server" Font-Bold="true"
ForeColor="Red"></asp:Label>
<table border="0" cellpadding="0" cellspacing="0" width="500">
<tr>
<td>
<asp:ImageButton ID="img_save" runat="server"
ImageUrl="~/icons/bullet_disk.png"
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
        onclick="img_save_Click" />
<asp:LinkButton ID="lnk_save" runat="server" Text="GUARDAR"
    onclick="lnk_save_Click"></asp:LinkButton>
<asp:ImageButton ID="img_regresar" runat="server"
    ImageUrl="~/icons/application_side_contract.png"
   PostBackUrl="~/Administrador/Usuario/Frm_RolLista.aspx" />
<asp:LinkButton ID="lnk_regresar" runat="server" Text="REGRESAR"

PostBackUrl="~/Administrador/Usuario/Frm_RolLista.aspx"></asp:LinkButton>
</td>
</tr>
</table>
<br />

<p class="style1"
    style="font-family:Comic Sans MS; font-size: xx-large; width:auto;">ROL DE
USUARIO</p>

DESCRIPCION:
<asp:TextBox ID="Txt_descripcion" runat="server"
Width="190px"></asp:TextBox>

</div>
</form>
</body>
</html>
```

Frm_Lista

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_RolLista.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Usuario.Frm_RolLista" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
<asp:Label ID="Label1" runat="server" Font-Bold="true" ForeColor="Red"
Visible="false"></asp:Label>
<asp:ImageButton ID="ImageButton1" runat="server"
    ImageUrl="~/icons/bug_add.png" onclick="ImageButton1_Click" />
<asp:LinkButton ID="Lnk_nuevo" runat="server"
onclick="Lnk_nuevo_Click">NUEVO</asp:LinkButton>
<br />
LISTA
DE&nbsp; TIPOS DE&nbsp; USUARIOS REGISTRADOS<br />
&nbsp;<asp:Label ID="lbl_mensaje" runat="server" Font-Bold="true"
ForeColor="Red"></asp:Label>
<br />
CRITERIO DE BUSQUEDA:
<asp:DropDownList ID="ddl_buscar" runat="server" Width="160 px">
<asp:ListItem Text="NOMBRE" value="n"></asp:ListItem>
</asp:DropDownList>
<asp:TextBox ID="txt_buscar" runat="server" Width="200
px"></asp:TextBox>
<asp:ImageButton ID="img_buscar" runat="server"
    ImageUrl="~/icons/application_form_magnify.png"
onclick="img_buscar_Click"/>
<asp:DataGrid ID="dgr_usuario" runat="server" CellPadding="4"
    ForeColor="#333333" GridLines="None" AutoGenerateColumns="False"
    onitemcommand="dgr_usuario_ItemCommand">
<AlternatingItemStyle BackColor="White" />
<EditItemStyle BackColor="#2461BF" />
<FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White"
/>
<ItemStyle BackColor="#EFF3FB" />
<PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
<SelectedItemStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
<Columns>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEditar" runat="server" CausesValidation="false"
CommandName="Editar"
    ImageUrl="~/icons/application_form_edit.png" ToolTip="Editar" Width="16"
/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn ItemStyle-Width="30 px">
```



```
<ItemTemplate>
  <asp:ImageButton ID="imgEliminar" runat="server" CausesValidation="false"
CommandName="Eliminar"
  ImageUrl="~/icons/award_star_delete.png" ToolTip="Eliminar" Width="16"
OnClientClick="return confirm('Seguro que desea eliminar el registro');" />
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="ID">
  <ItemTemplate>
    <span style="float:left;">
      <asp:Label ID="lbl_Idusuario" runat="server" Text='<#Eval("id_rol")
%>'></asp:Label>
    </ItemTemplate>
  </asp:TemplateColumn>
  <asp:TemplateColumn HeaderText="NOMBRE">
    <ItemTemplate>
      <span style="float:left;">
        <asp:Label ID="lbl_nombre" runat="server" Text='<#Eval("des_rol") %>'>
</asp:Label>
      </ItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn HeaderText="ESTADO">
      <ItemTemplate>
        <span style="float:left;">
          <asp:Label ID="lbl_estado" runat="server" Text='<#Eval("est_rol") %>'>
</asp:Label>
        </ItemTemplate>
      </asp:TemplateColumn>

    </Columns>
  </asp:DataGrid>

</div>
</form>
</body>
</html>
Frm_SocioNuevo

<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_SocioNuevo.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Socio.Frm_SocioNuevo" %>
<% @ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="asp" %>
```







</html>

Frm_SocioLista

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_SocioLista.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Socio.Frm_SocioLista" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:ImageButton ID="ImageButton1" runat="server"
                ImageUrl="~/icons/bug_add.png" onclick="ImageButton1_Click" />
            <asp:LinkButton ID="Lnk_nuevo" runat="server"
                onclick="Lnk_nuevo_Click">NUEVO</asp:LinkButton>
            <br />
            LISTA USUARIOS<br />
            &nbsp;<asp:Label ID="lbl_mensaje" runat="server" Font-Bold="true"
                ForeColor="Red"></asp:Label>
            <br />
            CRITERIO DE BUSQUEDA:
            <asp:DropDownList ID="ddl_buscar" runat="server" Width="160 px">
            <asp:ListItem Text="CEDULA" value="c"></asp:ListItem>
            <asp:ListItem Text="APELLIDO" Value="a"></asp:ListItem>
            </asp:DropDownList>
            <asp:TextBox ID="txt_buscar" runat="server" Width="200
                px"></asp:TextBox>
            <asp:ImageButton ID="img_buscar" runat="server"
                ImageUrl="~/icons/application_form_magnify.png"
                onclick="img_buscar_Click"/>
            <asp:DataGrid ID="dgr_usuario" runat="server" CellPadding="4"
                ForeColor="#333333" GridLines="None" AutoGenerateColumns="False"
                onitemcommand="dgr_usuario_ItemCommand">
            <AlternatingItemStyle BackColor="White" />
            <EditItemStyle BackColor="#2461BF" />
            <FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White"
/>
<ItemStyle BackColor="#EFF3FB" />
<PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
<SelectedItemStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
<Columns>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEditar" runat="server" CausesValidation="false"
CommandName="Editar"
ImageUrl="~/icons/application_form_edit.png" ToolTip="Editar" Width="16"
/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEliminar" runat="server" CausesValidation="false"
CommandName="Eliminar"
ImageUrl="~/icons/award_star_delete.png" ToolTip="Eliminar" Width="16"
OnClick="return confirm('Seguro que desea eliminar el registro')"/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="Id">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_Idusuario" runat="server" Text='<%#Eval("id_socio")
%>'></asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="CEDULA">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_cedula" runat="server" Text='<%#Eval("ced_socio") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="APELLIDO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_apellido1" runat="server" Text='<%#Eval("ape_socio")
%>'></asp:Label>
```




```
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="NOMBRE">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_nombre" runat="server" Text='<%#Eval("nom_socio")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="DIRECCION">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_estadocivil" runat="server" Text='<%#Eval("dir_socio")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="TELEFONO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_fecha" runat="server" Text='<%#Eval("tel_socio") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="CELULAR">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_cel" runat="server" Text='<%#Eval("cel_socio") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="GENERO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_genero" runat="server" Text='<%#Eval("gen_socio") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="FECHA DE NACIMIENTO">
<ItemTemplate>
<span style="float:left;">
```



```
<asp:Label ID="lbl_fen" runat="server" Text='<%#Eval("fen_socio") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="ESTADO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_est" runat="server" Text='<%#Eval("est_socio") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

</Columns>
</asp:DataGrid>

</div>
</form>
</body>
</html>
```

Frm_LoteNuevo

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_LoteNuevo.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Lote.Frm_LoteNuevo" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
<asp:Label ID="Lbl_mensaje" runat="server" Font-Bold="true"
ForeColor="Red"></asp:Label>
<table border="0" cellpadding="0" cellspacing="0" width="500">
<tr>
<td>
```

[illegible]



```
<br />
<br />
</div>
</form>
</body>
</html>
Frm_LoteLista

<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_LoteLista.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Lote.Frm_LoteLista" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:ImageButton ID="ImageButton1" runat="server"
        ImageUrl="~/icons/bug_add.png" onclick="ImageButton1_Click" />
      <asp:LinkButton ID="Lnk_nuevo" runat="server"
onclick="Lnk_nuevo_Click">NUEVO</asp:LinkButton>
      <br />
      LISTA DE LOTES REGISTRADOS<br />
      &nbsp;<asp:Label ID="lbl_mensaje" runat="server" Font-Bold="true"
ForeColor="Red"></asp:Label>
      <br />
      CRITERIO DE BUSQUEDA:
      <asp:DropDownList ID="ddl_buscar" runat="server" Width="160 px">
      <asp:ListItem Text="NUMERO" value="c"></asp:ListItem>
      <asp:ListItem Text="MANZANA" Value="a"></asp:ListItem>
      <asp:ListItem Text="SOCIO" Value="s"></asp:ListItem>
      </asp:DropDownList>
      <asp:TextBox ID="txt_buscar" runat="server" Width="200
px"></asp:TextBox>
      <asp:ImageButton ID="img_buscar" runat="server"
        ImageUrl="~/icons/application_form_magnify.png"
onclick="img_buscar_Click"/>
      <asp:DataGrid ID="dgr_usuario" runat="server" CellPadding="4"
        ForeColor="#333333" GridLines="None" AutoGenerateColumns="False"
```



```
        onitemcommand="dgr_usuario_ItemCommand">
        <AlternatingItemStyle BackColor="White" />
        <EditItemStyle BackColor="#2461BF" />
        <FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
        <HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White"
    />
        <ItemStyle BackColor="#EFF3FB" />
        <PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
        <SelectedItemStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
        <Columns>
        <asp:TemplateColumn ItemStyle-Width="30 px">
        <ItemTemplate>
        <asp:ImageButton ID="imgEditar" runat="server" CausesValidation="false"
CommandName="Editar"
        ImageUrl="~/icons/application_form_edit.png" ToolTip="Editar" Width="16"
    />
        </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn ItemStyle-Width="30 px">
        <ItemTemplate>
        <asp:ImageButton ID="imgEliminar" runat="server" CausesValidation="false"
CommandName="Eliminar"
        ImageUrl="~/icons/award_star_delete.png" ToolTip="Eliminar" Width="16"
OnClientClick="return confirm('Seguro que desea eliminar el registro')"/>
        </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn HeaderText="CODIGO">
        <ItemTemplate>
        <span style="float:left;">
        <asp:Label ID="lbl_Idusuario" runat="server" Text='<#Eval("id_lote")
%>'></asp:Label>
        </ItemTemplate>
        </asp:TemplateColumn>

        <asp:TemplateColumn HeaderText="NUMERO">
        <ItemTemplate>
        <span style="float:left;">
        <asp:Label ID="lbl_cedula" runat="server" Text='<#Eval("num_lote") %>'>
</asp:Label>
        </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn HeaderText="METROS">
```



```
<ItemTemplate>
  <span style="float:left;">
    <asp:Label ID="lbl_apellido1" runat="server" Text='<%#Eval("mtr_lote")
%>'> </asp:Label>
  </ItemTemplate>
</asp:TemplateColumn>

  <asp:TemplateColumn HeaderText="ESTADO">
    <ItemTemplate>
      <span style="float:left;">
        <asp:Label ID="lbl_nombre" runat="server" Text='<%#Eval("est_lote") %>'>
</asp:Label>
      </ItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn HeaderText="MANZANA">
      <ItemTemplate>
        <span style="float:left;">
          <asp:Label ID="lbl_estadocivil" runat="server" Text='<%#Eval("mnz_lote")
%>'> </asp:Label>
        </ItemTemplate>
      </asp:TemplateColumn>
      <asp:TemplateColumn HeaderText="SOCIO">
        <ItemTemplate>
          <span style="float:left;">
            <asp:Label ID="lbl_fecha" runat="server"
Text='<%#Eval("TBL_SOCIO.ape_socio") %>'> </asp:Label>
          </ItemTemplate>
        </asp:TemplateColumn>

      </Columns>
    </asp:DataGrid>

  </div>
</form>
</body>
</html>
```

Frm_TipoPagNuevo

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_TipoPagNuevo.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Pagos.Frm_TipoPagNuevo" %>
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:Label ID="Lbl_mensaje" runat="server" Font-Bold="true"
ForeColor="Red"></asp:Label>
      <table border="0" cellpadding="0" cellspacing="0" width="500">
<tr>
<td>
<asp:ImageButton ID="img_save" runat="server"
ImageUrl="~/icons/bullet_disk.png"
  onclick="img_save_Click" />
<asp:LinkButton ID="lnk_save" runat="server" Text="GUARDAR"
  onclick="lnk_save_Click"></asp:LinkButton>
<asp:ImageButton ID="img_regresar" runat="server"
  ImageUrl="~/icons/application_side_contract.png"
 PostBackUrl="~/Administrador/Pagos/Frm_TipoPagLista.aspx" />
<asp:LinkButton ID="lnk_regresar" runat="server" Text="REGRESAR"

PostBackUrl="~/Administrador/Pagos/Frm_TipoPagLista.aspx"></asp:LinkButton>
</td>
</tr>
</table>
<br />

    <p class="style1"
      style="font-family:Comic Sans MS; font-size: xx-large; width:auto;">TIPOS
DE ALICUOTAS</p>

    DESCRIPCION:
    <asp:TextBox ID="Txt_descripcion" runat="server"
Width="190px"></asp:TextBox>
  </div>
</form>
</body>
</html>
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



Frm_TipoPagLista

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_TipoPagLista.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Tipo_Doc.Frm_TipoDoc" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:ImageButton ID="ImageButton1" runat="server"
                ImageUrl="~/icons/bug_add.png" onclick="ImageButton1_Click" />
            <asp:LinkButton ID="Lnk_nuevo" runat="server"
                onclick="Lnk_nuevo_Click">NUEVO</asp:LinkButton>
            <br />
            LISTA TIPOS PAGOS<br />
            &nbsp;<asp:Label ID="lbl_mensaje" runat="server" Font-Bold="true"
                ForeColor="Red"></asp:Label>
            <br />
            CRITERIO DE BUSQUEDA:
            <asp:DropDownList ID="ddl_buscar" runat="server" Width="160 px">
            <asp:ListItem Text="NOMBRE" value="n"></asp:ListItem>
            </asp:DropDownList>
            <asp:TextBox ID="txt_buscar" runat="server" Width="200
                px"></asp:TextBox>
            <asp:ImageButton ID="img_buscar" runat="server"
                ImageUrl="~/icons/application_form_magnify.png"
                onclick="img_buscar_Click"/>
            <asp:DataGrid ID="dgr_usuario" runat="server" CellPadding="4"
                ForeColor="#333333" GridLines="None" AutoGenerateColumns="False"
                onitemcommand="dgr_usuario_ItemCommand">
                <AlternatingItemStyle BackColor="White" />
                <EditItemStyle BackColor="#2461BF" />
                <FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
                <HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White"
                />
                <ItemStyle BackColor="#EFF3FB" />
```




```
<PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
<SelectedItemStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
<Columns>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEditar" runat="server" CausesValidation="false"
CommandName="Editar"
ImageUrl="~/icons/application_form_edit.png" ToolTip="Editar" Width="16"
/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEliminar" runat="server" CausesValidation="false"
CommandName="Eliminar"
ImageUrl="~/icons/award_star_delete.png" ToolTip="Eliminar" Width="16"
OnClick="return confirm('Seguro que desea eliminar el registro')"/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="ID">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_Idtipousuario" runat="server" Text='<#Eval("id_tipo")
%>'></asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="NOMBRE">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_nombre" runat="server" Text='<#Eval("des_tipo") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="ESTADO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_estado" runat="server" Text='<#Eval("est_tipo") %>'>
</asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
</Columns>
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
</asp:DataGrid>

</div>
</form>
</body>
</html>
```

Frm_ActaNuevo

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_ActaNuevo.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Actas.Frm_ActaNuevo" %>
<% @ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="asp" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Label ID="Lbl_mensaje" runat="server" Font-Bold="true"
ForeColor="Red"></asp:Label>
            <asp:ToolkitScriptManager ID="ToolkitScriptManager1" runat="server">
                </asp:ToolkitScriptManager>
            <table border="0" cellpadding="0" cellspacing="0" width="500">
<tr>
<td>
<asp:ImageButton ID="img_save" runat="server"
ImageUrl="~/icons/bullet_disk.png"
onclick="img_save_Click" />
<asp:LinkButton ID="lnk_save" runat="server" Text="GUARDAR"
onclick="lnk_save_Click"></asp:LinkButton>
<asp:ImageButton ID="img_regresar" runat="server"
ImageUrl="~/icons/application_side_contract.png"
PostBackUrl="~/Administrador/Actas/Frm_ActaLista.aspx" />
<asp:LinkButton ID="lnk_regresar" runat="server" Text="REGRESAR"
PostBackUrl="~/Administrador/Actas/Frm_ActaLista.aspx"></asp:LinkButton>
</td>
</tr>
</td>
```

[illegible][illegible]

```
<asp:TextBox ID="Txt_fecha" runat="server" Width="118px"></asp:TextBox>
<asp:ImageButton ID="img_calendario" runat="server"
ImageUrl="~/icons/calendar.png" />
<asp:CalendarExtender ID="CalendarExtender" TargetControlID="Txt_fecha"
PopupButtonID="img_calendario"
runat="server" TodaysDateFormat="dd MMM , yyyy"
DaysModeTitleFormat="MMMM, yyyy" Format="dd/MM/yyyy">
```

[illegible]

```
<asp:DropDownList ID="ddl_usuario" runat="server" Width="161px"
Height="25px">
</asp:DropDownList>
```



</html>

Frm_ActaLista

```
<% @ Page Language="C#" AutoEventWireup="true"
CodeBehind="Frm_ActaLista.aspx.cs"
Inherits="SistemaPagoAlicuotas.Administrador.Actas.Frm_ActaLista" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:ImageButton ID="ImageButton1" runat="server"
                ImageUrl="~/icons/bug_add.png" onclick="ImageButton1_Click" />
            <asp:LinkButton ID="Lnk_nuevo" runat="server"
                onclick="Lnk_nuevo_Click">NUEVO</asp:LinkButton>
            <br />
            LISTA DE ACTAS<br />
            &nbsp;<asp:Label ID="lbl_mensaje" runat="server" Font-Bold="true"
                ForeColor="Red"></asp:Label>
            <br />
            CRITERIO DE BUSQUEDA:
            <asp:DropDownList ID="ddl_buscar" runat="server" Width="160 px">
            <asp:ListItem Text="OBSERVACION" value="c"></asp:ListItem>
            <asp:ListItem Text="APELLIDO" Value="a"></asp:ListItem>
            </asp:DropDownList>
            <asp:TextBox ID="txt_buscar" runat="server" Width="200
                px"></asp:TextBox>
            <asp:ImageButton ID="img_buscar" runat="server"
                ImageUrl="~/icons/application_form_magnify.png"
                onclick="img_buscar_Click"/>
            <asp:DataGrid ID="dgr_usuario" runat="server" CellPadding="4"
                ForeColor="#333333" GridLines="None" AutoGenerateColumns="False"
                onitemcommand="dgr_usuario_ItemCommand">
                <AlternatingItemStyle BackColor="White" />
                <EditItemStyle BackColor="#2461BF" />
                <FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
```

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



```
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White"
/>
<ItemStyle BackColor="#EFF3FB" />
<PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
<SelectedItemStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
<Columns>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEditar" runat="server" CausesValidation="false"
CommandName="Editar"
ImageUrl="~/icons/application_form_edit.png" ToolTip="Editar" Width="16"
/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="img_imprimir" runat="server"
CausesValidation="false"
CommandName="Imprimir" ImageUrl="~/icons/book.png"
ToolTip="Imprimir"
Width="16" />
</ItemTemplate>
<ItemStyle Width="30px" />
</asp:TemplateColumn>
<asp:TemplateColumn ItemStyle-Width="30 px">
<ItemTemplate>
<asp:ImageButton ID="imgEliminar" runat="server" CausesValidation="false"
CommandName="Eliminar"
ImageUrl="~/icons/award_star_delete.png" ToolTip="Eliminar" Width="16"
OnClick="return confirm('Seguro que desea eliminar el registro')"/>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="CODIGO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_Idusuario" runat="server" Text='<%#Eval("id_historial")
%>'></asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="FECHA">
<ItemTemplate>
```



```
<span style="float:left;">
<asp:Label ID="lbl_cedula" runat="server" Text='<%#Eval("fec_historial")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="OBSERVACION">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_nombre" runat="server" Text='<%#Eval("obs_historial")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

<asp:TemplateColumn HeaderText="USUARIO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_fecha" runat="server"
Text='<%#Eval("TBL_USUARIO.ape_usuario") %>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>
<asp:TemplateColumn HeaderText="ESTADO">
<ItemTemplate>
<span style="float:left;">
<asp:Label ID="lbl_genero" runat="server" Text='<%#Eval("est_historial")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateColumn>

</Columns>
</asp:DataGrid>

</div>
</form>
</body>
</html>
```

Manual de Usuario

Pantalla principal de acceso del sistema

COMITE PROMEJORAS SANTA ISABEL

Usuario

ALICIA

Contraseña

.....

Rol de usuario

ADMINISTRADOR ▼

INGRESAR SALIR

SISTEMA PAGO DE ALICUOTAS

Figura 77: Ingreso al sistema

Pasos:

1. Ingrese su usuario
2. Ingrese su contraseña

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



3. Ingrese su Rol (Secretaria, Administrador)
4. Si el usuario y contraseña es correcto accede al sistema caso contrario el usuario será incorrecto

Ventana Principal del Sistema



Figura 78: Ventana principal de acceso al sistema



Figura 79: Ingreso de nuevo socio

Pasos:

1. Seleccionamos la pestaña socios
2. Se nos abrirá un formulario
3. Llenamos el formulario
4. Hacemos clic en guardar
5. Y Tendremos al Socio Agregado

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

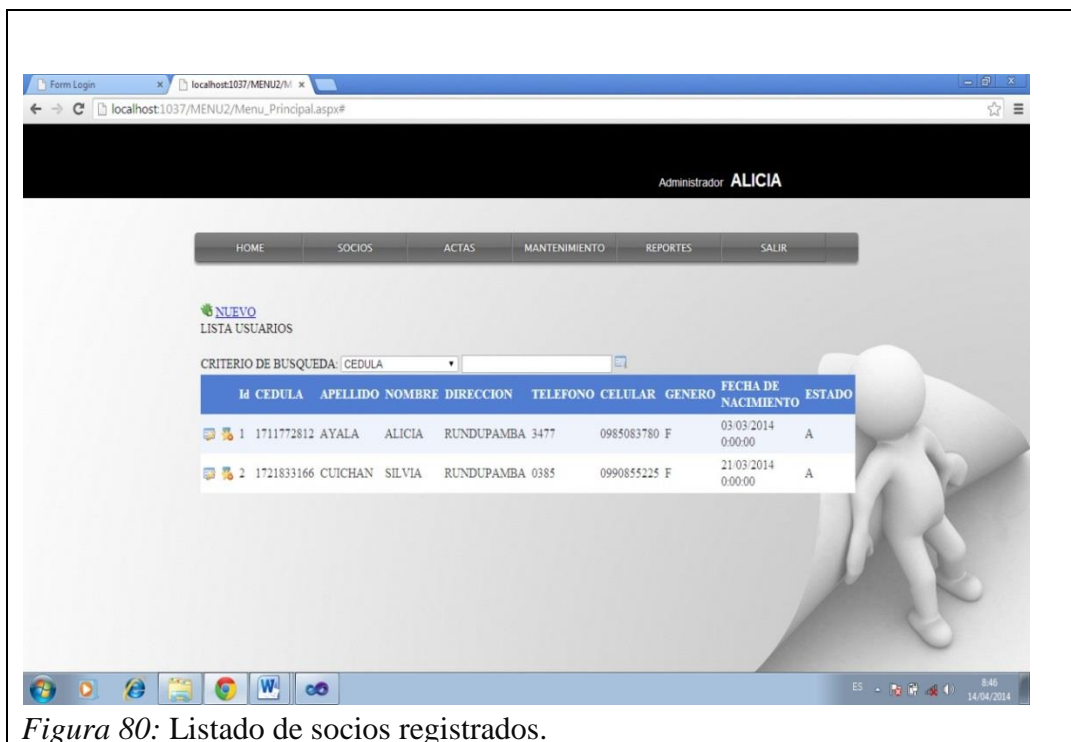


Figura 80: Listado de socios registrados.

Pasos:

1. Seleccionamos la pestaña socios
2. Seleccionamos regresar
3. Se nos abrirá el listado de los socios
4. Tenemos Acciones tales como Buscar, editar, eliminar

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**



Figura 81: En esta ventana podremos realizar actas

Pasos:

1. Seleccionamos la pestaña Actas
2. Se nos abrirá un formulario
3. Llenamos el formulario
4. Hacemos clic en guardar
5. Y Tendremos el acta guardada.

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

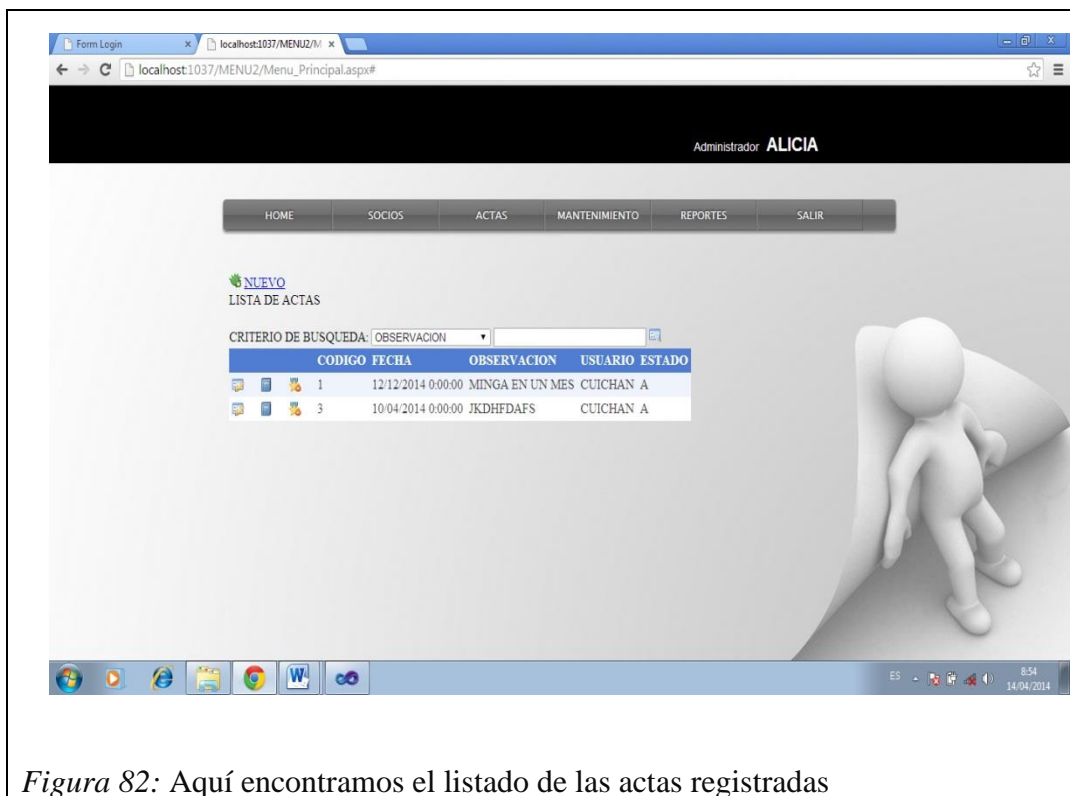


Figura 82: Aquí encontramos el listado de las actas registradas

Pasos:

1. Seleccionamos la pestaña Actas
2. Seleccionamos regresar
3. Se nos abrirá el listado de las actas registradas
4. Tenemos Acciones tales como Buscar, editar, eliminar.
5. También obtenemos un vínculo para poder imprimir el acta.

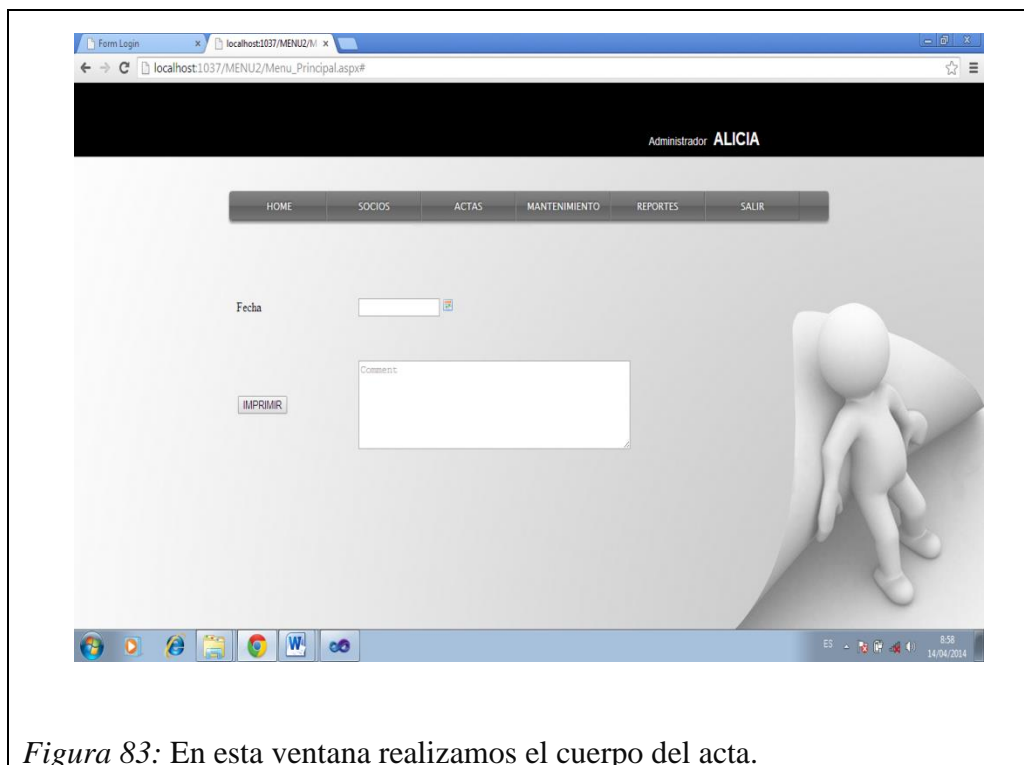


Figura 83: En esta ventana realizamos el cuerpo del acta.

Pasos:

1. Seleccionamos la pestaña Actas
2. Seleccionamos imprimir
3. Ingresamos fecha y escribimos el cuerpo del acta
4. Presionamos imprimir y se nos guarda un documento PDF.

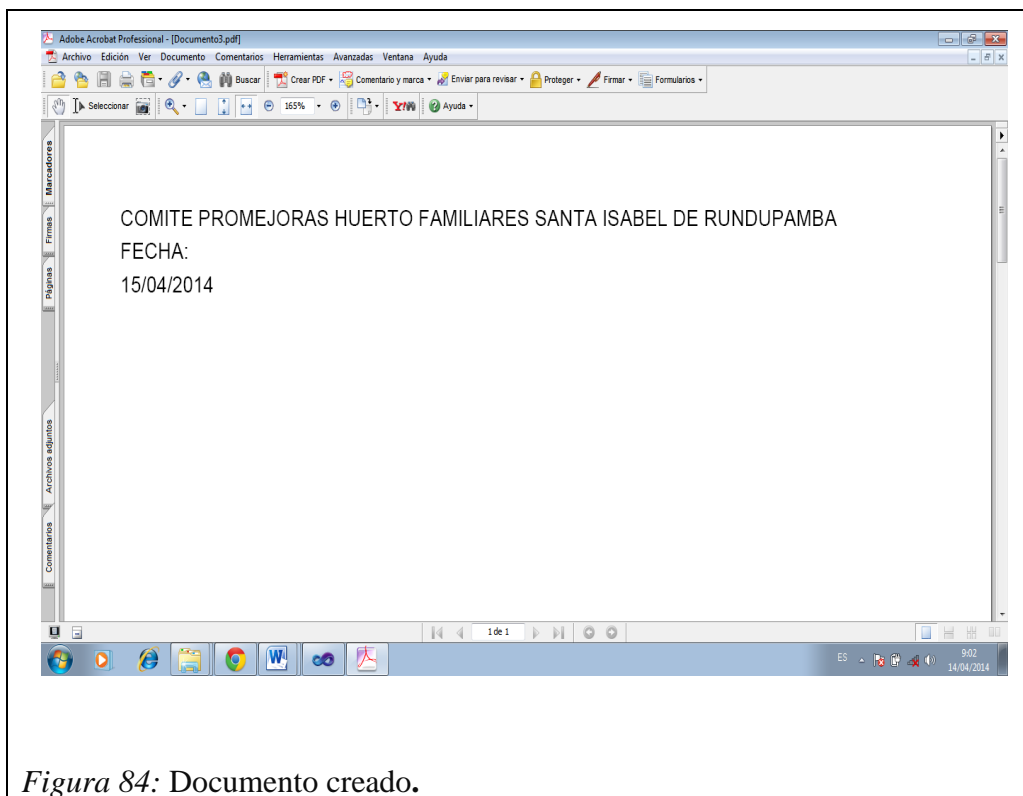


Figura 84: Documento creado.

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

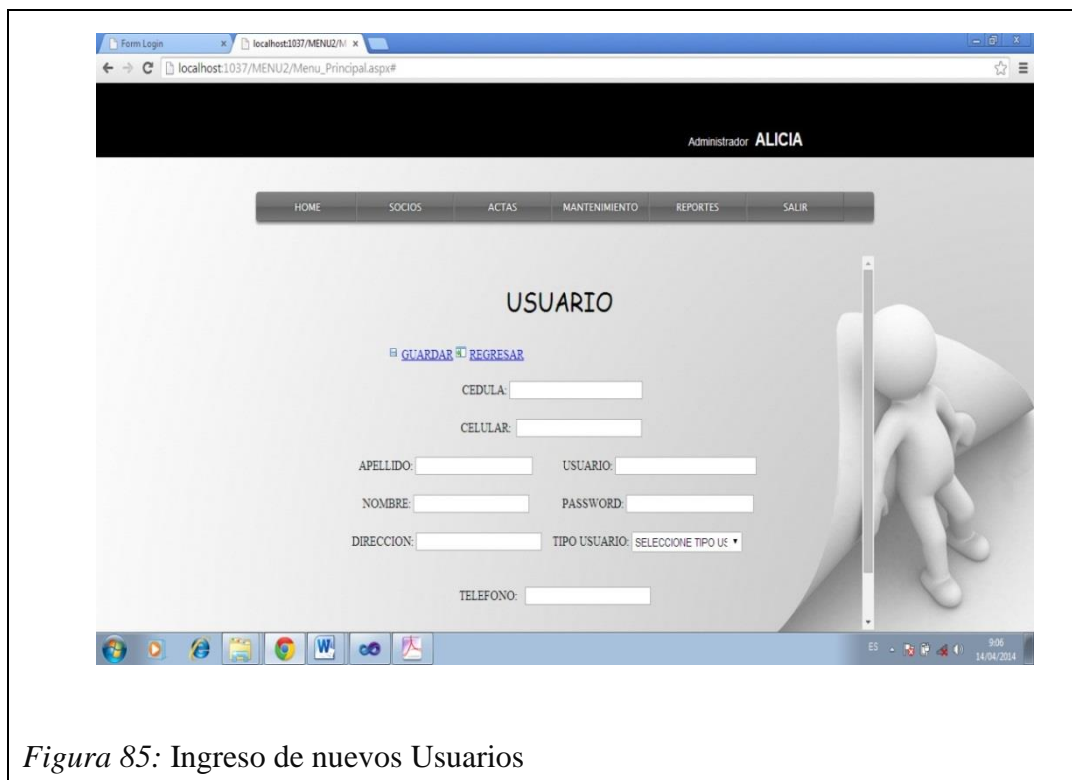


Figura 85: Ingreso de nuevos Usuarios

Pasos:

1. Seleccionamos la pestaña mantenimiento
2. A continuación la pestaña usuario
3. Llenamos el formulario
4. Hacemos clic en guardar
5. Y Tendremos al usuario guardado

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**

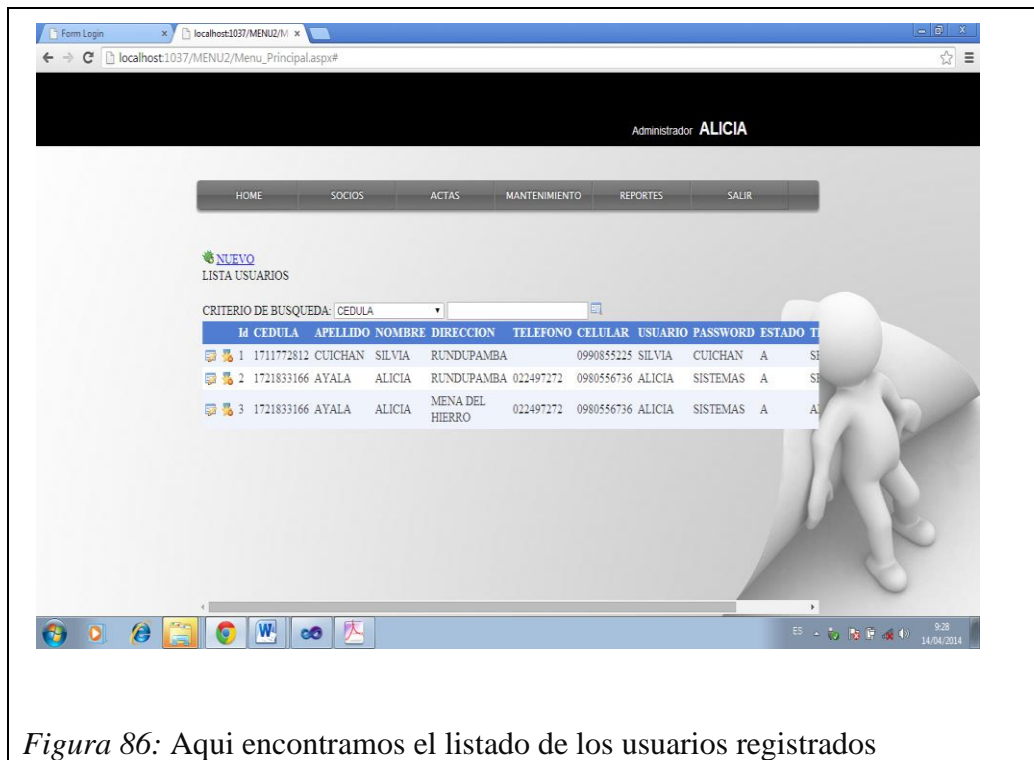


Figura 86: Aqui encontramos el listado de los usuarios registrados

1. Seleccionamos la pestaña usuarios
2. Seleccionamos regresar
3. Se nos abrirá el listado de los usuarios registradas
4. Tenemos Acciones tales como Buscar, editar, eliminar.



GLOSARIO

Actor.- Algo o alguien externo al sistema en desarrollo pero que interactúa con él.

Arquitectura.- Estructura lógica y física de un sistema empleado para diseñar todas las estrategias y tácticas aplicadas durante el desarrollo.

Atributo.- Definición de dato simple o compuesto perteneciente a un objeto de clase.

Clase.- Descripción de un grupo de objetos con atributos, conducta y relaciones comunes.

Caso de uso.- Representación de un proceso del negocio. representa el modelo de diálogo entre un actor y el sistema

Diagrama de casos de uso.- Representación gráfica que representa algunos o todos los actores, casos de uso y sus interacciones en el sistema.

Diagrama de clases.- Representación gráfica que permite visualizar algunas o todas las clases de un modelo

Diagrama de secuencias.- Representación gráfica que describe interacciones de secuencia de objetos.

UML (lenguaje de modelamiento unificado).- Lenguaje usado para especificar, visualizar y documentar un sistema en desarrollo orientado a objetos.

Base de datos.- Estructura de software que colecciona información muy variada de diferentes personas y cosas (es decir, de una realidad determinada), cada una de las cuales tiene algo en común o campos comunes con todos o con algunos. Se diseñó



con la finalidad de solucionar y agilizar la administración de los datos que se almacenan en la memoria del computador.

Hardware.- Todos aquellos componentes físicos de un computador, todo lo visible y tangible. Por extensión, se aplica también a otros componentes electrónicos que no necesariamente forman parte de un computador.

WEBGRAFIA

<http://alejandroesteban.files.wordpress.com/2008/09/el-proceso-de-instalacion-de-microsoft-sql-server-2008.pdf>

<http://www.slideshare.net/SonGoku10/manual-de-instalacin-sql-server-2008>

<http://www.ajpdsoft.com/modules.php?name=News&file=article&sid=517>

[http://www.wisis.ufg.edu.sv/AB63CA21-1D99-49FD-9A43-](http://www.wisis.ufg.edu.sv/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/FinalDownload/DownloadId-262810B4356D1BF1BB0179DB65BE9F69/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/www.wisis/documentos/TE/338.4791-A473d/338.4791-A473d-Anexo%207.pdf)

[30AD0BD318AA/FinalDownload/DownloadId-](http://www.wisis.ufg.edu.sv/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/FinalDownload/DownloadId-262810B4356D1BF1BB0179DB65BE9F69/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/www.wisis/documentos/TE/338.4791-A473d/338.4791-A473d-Anexo%207.pdf)

[262810B4356D1BF1BB0179DB65BE9F69/AB63CA21-1D99-49FD-9A43-](http://www.wisis.ufg.edu.sv/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/FinalDownload/DownloadId-262810B4356D1BF1BB0179DB65BE9F69/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/www.wisis/documentos/TE/338.4791-A473d/338.4791-A473d-Anexo%207.pdf)

[30AD0BD318AA/www.wisis/documentos/TE/338.4791-A473d/338.4791-A473d-](http://www.wisis.ufg.edu.sv/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/FinalDownload/DownloadId-262810B4356D1BF1BB0179DB65BE9F69/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/www.wisis/documentos/TE/338.4791-A473d/338.4791-A473d-Anexo%207.pdf)
[Anexo%207.pdf](http://www.wisis.ufg.edu.sv/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/FinalDownload/DownloadId-262810B4356D1BF1BB0179DB65BE9F69/AB63CA21-1D99-49FD-9A43-30AD0BD318AA/www.wisis/documentos/TE/338.4791-A473d/338.4791-A473d-Anexo%207.pdf)

http://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso

<http://docs.kde.org/stable/es/kdesdk/umbrello/uml-elements.html>

[http://www.estrategiasdeinversion.com/noticias/20120705/son-indicadores-para-](http://www.estrategiasdeinversion.com/noticias/20120705/son-indicadores-para-sirven)
[sirven](http://www.estrategiasdeinversion.com/noticias/20120705/son-indicadores-para-sirven)

<http://www.iadb.org/ove/spbook/lamatriz.htm>

**REGISTRO Y CONTROL DE INFORMACION PERSONAL Y PAGO DE
ALICUOTAS MEDIANTE UN SISTEMA INFORMATICO WEB PARA EL
COMITÉ PRO-MEJORAS SANTA ISABEL**