

UNIVERSIDAD POLITECNICA SALESIANA

CARRERA DE INGENIERIA DE SISTEMAS

SEDE CUENCA

TEMA:

Prototipo de un sistema de prevención, control, y corrección del uso inadecuado de las bocinas de los automotores, a fin de disminuir la contaminación ambiental sonora, producido por las unidades de transporte público en la ciudad de Cuenca

TRABAJO PREVIO A LA
OBTENCION DEL TITULO DE
INGENIERO DE SISTEMAS

AUTOR:

DAVID HIPOLITO MOGROVEJO PENALOZA

DIRECTORA:

ING. BERTHA TACURI CAPELO

CUENCA - ECUADOR

2012

Declaratoria de responsabilidad

Los conceptos desarrollados, análisis realizados y las conclusiones del presente trabajo, son de exclusiva responsabilidad del autor.

A través de la presente declaración cedo los derechos de propiedad intelectual correspondiente a este trabajo a la Universidad Politécnica Salesiana, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

Cuenca, mayo de 2012

A handwritten signature in blue ink, appearing to be 'D. Mogrovejo Peñalosa', written in a cursive style.


DAVID HIPOLITO MOGROVEJO PEÑALOZA
AUTOR

Ing. Bertha Tacuri

CERTIFICA:

Haber dirigido y revisado el informe de monografía realizado por el Señor David Hipólito Mogrovejo Peñaloza, así como, el cumplimiento y desarrollo de la parte práctica de la misma, en base a ello y cumpliendo con todos los requisitos necesarios, autorizo la presentación de la misma

Cuenca Mayo del 2012

A handwritten signature in blue ink, appearing to read 'Bertha Tacuri', enclosed within a blue oval scribble.

Ing. Bertha Tacuri
DIRECTORA

Tabla de contenido

1. CONTAMINACIÓN AUDITIVA.....	1
1.1. Introducción	2
1.2. Causas.....	2
1.3. Consecuencias.....	3
1.3.1. Efectos sobre la persona	3
Existen varios efectos que produce el ruido en las personas, como se muestra en la <i>Figura1</i>	3
1.4. ESTADÍSTICAS	9
1.4.1. Resumen de Valores Críticos.....	9
A partir de los valores indicados en la primera columna de la <i>Tabla 1</i> se empiezan a sentir, dependiendo de la sensibilidad individual, los efectos señalados en la segunda.....	9
2. SOFTWARE Y ENTORNOS DE DESARROLLO.....	12
2.1. Introducción	13
2.2. TECNOLOGÍAS PARA EL DESARROLLO DE APLICACIONES	13
2.2.1. Arquitectura de Software.....	13
2.2.2. Escalabilidad	16
2.2.3. Integridad	17
2.2.4. Disponibilidad.....	17
2.3. ELECCION DE LA TECNOLOGIA	18
2.3.1. Open Source Vs SoftwarePropietario.....	18
2.3.2. Entornos de desarrollo (IDE's).....	19
2.3.3. Comparación funcional	20
2.4. APRENDIZAJE DE LA TECNOLOGÍA ESCOGIDA.....	21
2.4.1. Componentes del entorno de desarrollo	21
2.4.2. Aplicación de ejemplo	22
2.4.3. Empezando la aplicación.....	24
2.4.4. Escribir el código	26
3. ESTRUCTURA Y DISEÑO DEL SISTEMA DE CONTROL SONORO.....	30
3.1. Introducción	31
3.2. AMBITO DEL PROYECTO:.....	31
3.2.1. Hardware.....	31

3.2.2.	Características físicas de dispositivo:	33
3.2.3.	Software	35
3.2.4.	GESTION DE COOPERATIVAS:.....	35
3.2.5.	<i>GESTION DE BUSES:</i>	37
3.2.6.	GESTION CHOFERES	37
3.2.7.	GESTION DE USO DE BOCINAS	39
3.2.8.	GESTION DE DISPOSITIVOS.....	39
3.2.9.	GESTION DE USUARIOS	40
3.2.10.	GESTION DE RUTAS	41
3.2.11.	GESTION DE REPRESENTANTES.....	42
3.3.	CASOS DE USO DEL SISTEMA DE CONTROL SONORO	44
3.3.1.	USUARIOS.....	44
3.3.2.	COOPERRATIVAS	46
3.3.3.	BUSES	50
3.3.4.	RUTAS.....	52
3.3.5.	CHOFERES.....	55
4.	PROGRAMACION E IMPLEMENTACION.....	58
4.1.	Introducción	59
4.2.	FUNDAMENTOS DEL LEGUAJE DE PROGRAMACION.....	59
4.2.1.	Microsoft Visual Studio	59
4.2.2.	ADO Entity Framework.....	60
4.3.	Desarrollo del Sistema de control sonoro.....	67
4.4.	Base de datos:	68
4.4.1.	Diagrama Entidad Relación:	68
4.5.	Mantenimiento de usuario:	69
5.	Anexos, conclusiones y recomendaciones	79
I.	Discusión.	80
II.	Conclusiones.....	81
III.	Recomendaciones.	82

El sistema de control sonoro únicamente se trata de una sistema que pretende controlar, el uso inadecuado de las bocinas, por lo cual se recomienda hacer campañas de concientización ya que el principal de problema para la contaminación auditiva no es más que un particular cultural, por ende se puede remediar con mucha información y campañas de sanciones..... 82

Con lo cual el sistema necesita trabajar mancomunadamente con las ordenanzas municipales así como la Jefatura de tránsito, a más de una ley que regule el uso de las bocinas.	82
De esta manera el sistema se volverá más eficiente y efectivo siendo así una buena opción para evitar en gran parte la contaminación auditiva.	82
IV. Bibliografía.	83
V. Anexos.	84
Script de creación de la base de datos.	84

Tabla de Ilustraciones

Figura 1 Efectos del ruido sobre las personas.	3
Figura 2 Pantalla Principal del entorno de desarrollo MS Visual Studio .NET 2008	21
Figura 3 Crear una aplicación de ejemplo.	22
Figura 4 Panel Explorador de proyectos MS Visual Studio	23
Figura 5 Pantalla completa Proyecto de prueba.	23
Figura 6 Formulario del proyecto de prueba Campo de texto.	24
Figura 7 Formulario del proyecto de prueba Botón.	24
Figura 8 Panel de Propiedades del campo del botón.	25
Figura 9 Panel de Propiedades del Label	25
Figura 10 Parte Superior del Panel de propiedades.	26
Figura 11 Porción de Código (Button)	26
Figura 12 Propiedad de Eventos del Botón	27
Figura 13 Explorador de soluciones Ver código	27
Figura 14 Navegador de Objetos dentro del código	28
Figura 15 Listado de eventos del objeto seleccionado	28
Figura 16 Código fuente botón	28
Figura 17 Porción de código de prueba	29
Figura 18 Programa de Prueba en tiempo de ejecución	29
Figura 19 Inclusión de Botón de Cerrar.	¡Error! Marcador no definido.
Figura 20 Porción de código Botón cerrar.	¡Error! Marcador no definido.
Figura 56 Vista aérea del dispositivo electrónico.	33
Figura 57 Perspectiva frontal del dispositivo electrónico	33
Figura 58 Esquema del Circuito eléctrico del dispositivo de adquisición de sonido.	34
Figura 59 Diagrama de clases del SCS.	43
Figura 60 Actores Identificados CU. Usuarios	44
Figura 61 Caso de Uso Usuarios	46

Figura62 Actores identificados en el caso de Uso Cooperativas.....	46
Figura 63 Diagrama de caso de uso de las cooperativas.....	49
Figura 64 Actores Caso de Uso de Buses.....	50
Figura 65 Diagrama de caso de uso de Buses	52
Figura 66 Actores Caso de Uso de Rutas.....	52
Figura 67 Diagrama de caso de uso de Rutas.....	55
Figura 68 Actores Caso de Uso de Choferes	55
Figura 69 Caso de uso de choferes.....	57
Figura 70 Ingreso a Visual Studio	61
Figura 71 Creación de una aplicación de prueba	61
Figura 72 Agregar un nuevo elemento.....	62
Figura73 ADO.NET Entity Data Model.....	62
Figura 74 Creación del modelo a partir de una base de datos existente.....	63
Figura 75 Selección de la tabla usuarios	64
Figura 76 Modelo de la tabla Usuarios	65
Figura 77 Creación del formulario de prueba dos campos	65
Figura 78 Código para manejo de datos con EF y Visual Basic .NET	66
Figura 79 Aplicación de prueba en ejecución	66
Figura 80 Diagrama de Entidades dentro de Visual Studio.....	67
Figura 81 Modelo de la base de datos para el SCS	68
Figura83 Microsoft SQL server Management Studio	76
Figura 84 Crear copia de seguridad.....	76
Figura 85 Ubicación de guardado de la base de datos.....	77
Figura 86 Mensaje de creación de respaldo de Base de datos	77
Figura 87 Opción restaurar base de datos	78
Figura 88 Selección de backup a restaurar	78

Contenido de Tablas

Tabla 1 Resumen de valores críticos del ruido.....	9
Tabla 2Detalle de los valores de afecciones y molestias en decibeles.....	10
Tabla 3Algunos ejemplos de lenguajes ampliamente usados.....	16

CAPITULO I

1. CONTAMINACIÓN AUDITIVA

1.1. Introducción

El constante crecimiento de las principales ciudades del Ecuador en los últimos años, han sido los causantes de un incremento en el tráfico vehicular, mismo que se ha considerado como el factor principal de la contaminación auditiva, entiéndase contaminación auditiva como ruido o sonido desagradable y/o molesto al oído, el cual se produce por uso de las bocinas de los automotores, y que se ha convertido no solo en uno de los principales contaminantes del medio ambiente, sino que sus efectos desencadenan en afecciones de salud. Desde hace ya varios años se ha considerado el hecho de tener una legislación que establezca una normativa de aplicación, obligatoria en relación con el campo de la acústica

Las leyes y reglamentos referentes al control del ruido son mucho más rigurosos en los países desarrollados. Sus habitantes han aprendido a evitar el ruido porque saben que éste les produciría estrés y trastornos en su salud física y mental, lo que afectaría a su calidad de vida. Por eso, en esos países no admiten el ruido provocado por motores, pitos y alta voces de vehículos; ni alto volumen en equipos de sonido; porque saben respetar la tranquilidad y el descanso de las demás personas. Incluso los propietarios de discotecas deben instalar ventanas sonoamortiguadas de doble o triple vidrio en sus locales, para obtener un permiso de funcionamiento de ahí que los países industrializados registren menores niveles de ruido que los que están en vías de desarrollo.

1.2. Causas

Los principales causantes de ruido en una ciudad son:

- a) El tránsito vehicular: los autos, camiones, motos. son los elementos componentes del ruido producido por el tránsito vehicular. Este ruido a su vez contiene otros elementos causantes del ruido como por ejemplo: el rugido de los motores, el uso inadecuado de las bocinas, los escapes defectuosos y a veces con resonadores intencionalmente adheridos en los escapes para incrementar el ruido.
- b) Aeropuertos
- c) Comercio e industrias
- d) Las discotecas, los estadios deportivos, espectáculos al aire libre.

1.3. Consecuencias

El ruido actúa a través del órgano del oído sobre los sistemas nerviosos central y autónomo¹. Cuando el estímulo sobrepasa determinados límites, se produce sordera y efectos patológicos en ambos sistemas, tanto instantáneos como diferidos. A niveles mucho menores, el ruido produce malestar y dificultad para escuchar e impide la atención, la comunicación, la concentración, el descanso y el sueño.

La reiteración de estas situaciones puede ocasionar estados crónicos de nerviosismo y estrés lo que, a su vez, lleva a trastornos psicofísicos, enfermedades cardiovasculares y alteraciones del sistema inmunitario.

La disminución del rendimiento escolar o profesional, los accidentes laborales o de tráfico, ciertas conductas antisociales, la tendencia al abandono de las ciudades, la pérdida de valor de los inmuebles entre otras tantas más son las consecuencias del sometimiento al ruido.

1.3.1. Efectos sobre la persona

Existen varios efectos que produce el ruido en las personas, como se muestra en la *Figura 1*

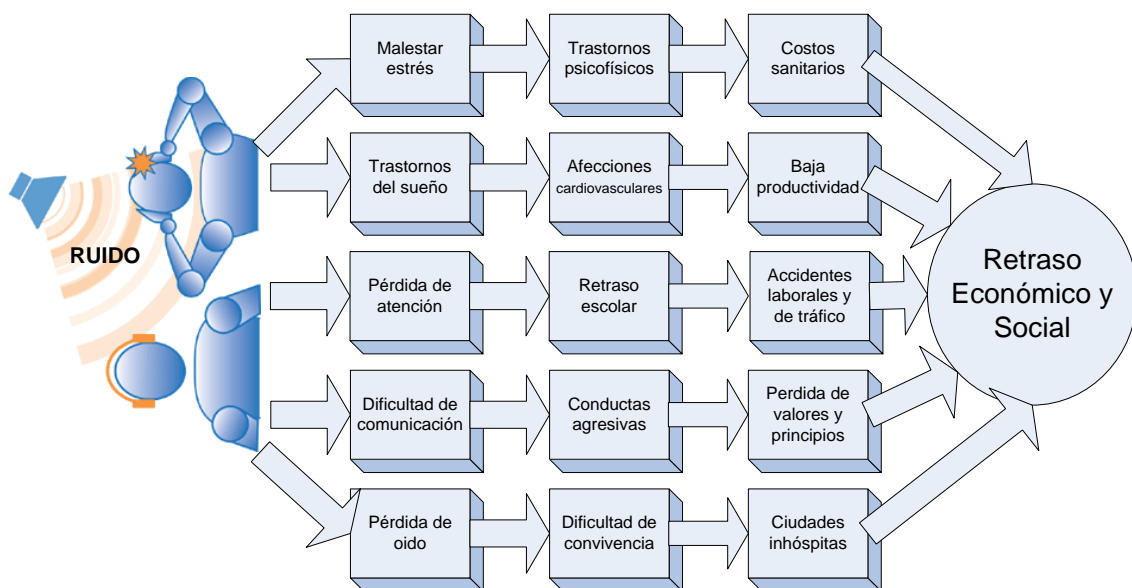


Figura 1 Efectos del ruido sobre las personas

¹Dr. Pedro Barreda. Publicado Medline revisión 2007.

²dB Decibelio es la unidad relativa empleada en acústica y telecomunicaciones para expresar la relación

1.3.1.1. Malestar

Este es quizás el efecto más común del ruido sobre las personas y la causa inmediata de la mayor parte de las quejas.

La sensación de malestar procede no sólo de la interferencia con la actividad en curso o con el reposo sino también de otras sensaciones, menos definidas pero a veces muy intensa, de estar siendo perturbado. Las personas afectadas hablan de intranquilidad, inquietud, desasosiego, depresión, desamparo, ansiedad o rabia. Todo ello contrasta con la definición de "salud" dada por la Organización Mundial de la Salud: "*Un estado de completo bienestar físico, mental y social, no la mera ausencia de enfermedad*".

El nivel de malestar varía no solamente en función de la intensidad del ruido y de otras características físicas del mismo que son menos objetivas (ruidos "chirriantes", "estridentes", etc.) sino también de factores tales como miedos asociados a la fuente del ruido, o el grado de legitimación que el afectado atribuya a la misma. Si el ruido es intermitente influyen también la intensidad máxima de cada episodio y el número de éstos.

Durante el día se suele experimentar malestar moderado a partir de los 50 decibelios (dB²), y fuerte a partir de los 55 dB. En el periodo vespertino, en estado de vigilia, estas cifras disminuyen en 5 ó 10 decibelios.

1.3.1.2. Interferencia con la comunicación

El nivel del sonido de una conversación en tono normal es, a un metro del hablante, de entre 50 y 55 dB. Hablando a gritos se puede llegar a 75 u 80 dB. Por otra parte, para que la palabra sea perfectamente inteligible es necesario que su intensidad supere en alrededor de 15 dB al ruido de fondo.

²dB **Decibelio** es la unidad relativa empleada en acústica y telecomunicaciones para expresar la relación entre dos magnitudes, acústicas o eléctricas, o entre la magnitud que se estudia y una magnitud de referencia.

Por lo tanto, un ruido superior a 35 ó 40 decibelios provocará dificultades en la comunicación oral que sólo podrán resolverse, parcialmente, elevando el tono de voz. A partir de 65 dB de ruido, la conversación se torna extremadamente difícil.

Situaciones parecidas se dan cuando el sujeto está intentando escuchar otras fuentes de sonido (televisión, música, etc.). Ante la interferencia de un ruido, se reacciona elevando el volumen de la fuente creándose así una mayor contaminación sonora sin lograr totalmente el efecto deseado.

1.3.1.3. Pérdida de atención, de concentración y de rendimiento

Es evidente que cuando realizamos alguna tarea necesita la utilización de señales acústicas, el ruido de fondo puede enmascarar estas señales o interferir con su percepción. Por otra parte, un ruido repentino producirá distracciones que reducirán el rendimiento en muchos tipos de trabajos, especialmente en aquellos que exijan un cierto nivel de concentración.

En ambos casos se afectará la realización de la tarea, apareciendo errores y disminuyendo la calidad y cantidad del producto de la misma.

Algunos accidentes, tanto laborales como de circulación, pueden ser debidos a este efecto.

En ciertos casos las consecuencias serán duraderas, por ejemplo, los niños sometidos a altos niveles de ruido durante su edad escolar no sólo aprenden a leer con mayor dificultad sino que también tienden a alcanzar grados inferiores de dominio de la lectura.

1.3.1.4. Trastornos del sueño

El ruido influye negativamente sobre el sueño de tres formas diferentes que se dan, en mayor o menor grado según peculiaridades individuales, a partir de los 30 decibelios:

- a) Mediante la dificultad o imposibilidad de dormirse.
- b) Causando interrupciones del sueño que, si son repetidas, pueden llevar al insomnio. La probabilidad de despertar depende no solamente de la intensidad del suceso ruidoso sino también de la

diferencia entre ésta y el nivel previo de ruido estable. A partir de 45 dB la probabilidad de despertar es grande.

- c) Disminuyendo la calidad del sueño, volviéndose éste menos tranquilo y acortándose sus fases más profundas, tanto las de sueño paradójico (los sueños) como las no-paradójicas. Aumentan la presión arterial y el ritmo cardiaco, hay vasoconstricción y cambios en la respiración.

Como consecuencia de todo ello, la persona no habrá descansado bien y será incapaz de realizar adecuadamente al día siguiente sus tareas cotidianas. Si la situación se prolonga, el equilibrio físico y psicológico se ven seriamente afectados.

Con frecuencia se intenta evitar o, al menos paliar, estas situaciones mediante la ingestión de tranquilizantes, el uso de tapones auditivos o cerrando las ventanas para dormir. Las dos primeras prácticas son, evidentemente, poco saludables por no ser naturales y poder acarrear dependencias y molestias adicionales. La tercera hace también perder calidad al sueño por desarrollarse éste en un ambiente mal ventilado y/o con una temperatura demasiado elevada.

1.3.1.5. Daños al oído

El efecto descrito en este apartado (pérdida de capacidad auditiva) no depende de la cualidad más o menos agradable que se atribuya al sonido percibido ni de que éste sea deseado o no. Se trata de un efecto físico que depende únicamente de la intensidad del sonido, aunque sujeto naturalmente a variaciones individuales.

- En la sordera transitoria o fatiga auditiva no hay aún lesión. La recuperación es normalmente casi completa al cabo de dos horas y completa a las 16 horas de cesar el ruido, si se permanece en un estado de confort acústico (menos de 50 decibelios en vigilia o de 30 durante el sueño).
- La sordera permanente está producida, bien por exposiciones prolongadas a niveles superiores a 75 dB, bien por sonidos de corta duración de más de 110 dB, o bien por acumulación de fatiga auditiva sin tiempo suficiente de recuperación. Hay lesión del oído

interno (células ciliadas externas de la superficie vestibular³ y de las de sostén de Deiters). Se produce inicialmente en frecuencias no conversacionales, por lo que el sujeto no la suele advertir hasta que es demasiado tarde, salvo casos excepcionales de auto observación. Puede ir acompañada de zumbidos de oído (acúfenos⁴) y de trastornos del equilibrio (vértigos).

1.3.1.6.El estrés: manifestaciones y consecuencias

Las personas sometidas de forma prolongada a situaciones como las anteriormente descritas (ruidos que hayan perturbado y frustrado sus esfuerzos de atención, concentración o comunicación, o que hayan afectado a su tranquilidad, su descanso o su sueño) suelen desarrollar algunos de los síndromes siguientes:

- Cansancio crónico
- Tendencia al insomnio, con la consiguiente agravación de la situación.
- Enfermedades cardiovasculares: hipertensión, cambios en la composición química de la sangre, isquemias cardiacas, etc. Se han mencionado aumentos de hasta el 20% o el 30% en el riesgo de ataques al corazón en personas sometidas a más de 65 decibelios en periodo diurno.
- Trastornos del sistema inmune responsable de la respuesta a las infecciones y a los tumores.
- Trastornos psicofísicos tales como ansiedad, manía, depresión, irritabilidad, náuseas, jaquecas, y neurosis o psicosis en personas predispuestas a ello.
- Cambios conductuales, especialmente comportamientos antisociales tales como hostilidad, intolerancia, agresividad,

³Sistema vestibular dentro del oído interno el sistema vestibular o aparato vestibular está formado por el utrículo, el sáculo y tres canales semicirculares (anterior, posterior y lateral). Cada una de estas estructuras contiene células especializadas para detectar aceleración y desaceleración, ya sea lineal (como es el caso de la mácula y el utrículo) o angular (canales semicirculares).

La función de este receptor es la mantención del equilibrio. Entre los receptores y emisores de información al cerebro.

⁴ El tinnitus o acúfenos es un fenómeno perceptivo que consiste en notar golpes o pitidos en el oído, que no proceden de ninguna fuente externa

aislamiento social y disminución de la tendencia natural hacia la ayuda mutua.

1.3.1.7. Grupos especialmente vulnerables

Ciertos grupos son especialmente sensibles al ruido. Entre ellos se encuentran los niños, los ancianos, los enfermos, las personas con dificultades auditivas o de visión y los fetos. Estos grupos tienden, por razones de comodidad, a estar sobre presentados en las muestras de las investigaciones en las que se basa la normativa sobre ruidos por lo que muchas veces se minusvaloran sus necesidades de protección.

La habituación al ruido

Se han citado casos de soldados que han podido dormir junto a una pieza de artillería que no cesaba de disparar o de comunidades que, a pesar de la cercanía de un aeropuerto, logran conciliar el sueño, aun cuando éste sea de poca calidad. Es cierto que a medio o largo plazo el organismo se habitúa al ruido, empleando para ello dos mecanismos diferentes por cada uno de los cuales se paga un precio distinto.

El primer mecanismo es la disminución de la sensibilidad del oído y su precio, la sordera temporal o permanente. Muchas de las personas a las que el ruido no molesta dirían, si lo supiesen, que no oyen el ruido o que lo oyen menos que otros o menos que antes. Naturalmente tampoco oyen otros sonidos que les son necesarios.

Mediante el segundo mecanismo, son las capas corticales del cerebro las que se habitúan. Dicho de otra forma, oímos el ruido pero no nos damos cuenta. Durante el sueño, las señales llegan a nuestro sistema nervioso, no nos despiertan pero desencadenan consecuencias fisiológicas de las que no somos conscientes: frecuencia cardiaca, flujo sanguíneo o actividad eléctrica cerebral. Es el llamado síndrome de adaptación.

1.3.1.8. Otros efectos

Sociales y económicos

La combinación de todos los factores anteriormente descritos ha convertido en inhóspitas muchas ciudades, deteriorando en ellas fuertemente los niveles de comunicación y las pautas de convivencia. En consecuencia, un número creciente de ciudadanos ha fijado su residencia en lugares inicialmente más sosegados.

1.4. ESTADÍSTICAS

1.4.1. Resumen de Valores Críticos

A partir de los valores indicados en la primera columna de la *Tabla 1* se empiezan a sentir, dependiendo de la sensibilidad individual, los efectos señalados en la segunda.

Tabla 1 Resumen de valores críticos del ruido⁵

A partir de este valor en dB	Se empiezan a sentir estos efectos nocivos
30	Dificultad en conciliar el sueño
30	Pérdida de calidad del sueño
40	Dificultad en la comunicación verbal
45	Probable interrupción del sueño
50	Malestar diurno moderado
55	Malestar diurno fuerte
65	Comunicación verbal extremadamente difícil
75	Pérdida de oído a largo plazo
110 - 140	Pérdida de oído a corto plazo

⁵Fuente: Armstrong T: Cumulative trauma disorders of the hand and wrist. AIHA ergonomics guide. AIHA 1994

Tabla 2Detalle de los valores de afecciones y molestias en decibeles⁶

Recinto	Valores	Límite	Apto	
	Efectos en la salud	LAeq (dB)	Tiempo (horas)	LAmaz, fast (dB)
Exterior habitable	Malestar fuerte, día y anochecer	55	16	-
	Malestar moderado, día y anochecer	50	16	-
Interior de viviendas	Interferencia en la comunicación verbal, día y anochecer	35	16	
	Perturbación del sueño, noche			
Dormitorios		30	8	45
Fuera de los dormitorios	Perturbación del sueño, ventana abierta (valores en el exterior)	45	8	60
Aulas de escolar y preescolar, interior	Interferencia en la comunicación, perturbación en la extracción de información, inteligibilidad del mensaje	35	Durante la clase	-
Dormitorios de preescolar, interior	Perturbación del sueño	30	Horas de descanso	45
Escolar, terrenos de juego	Malestar (fuentes externas)	55	Durante el juego	-
Salas de hospitales, interior	Perturbación del sueño, noche	30	8	40
	Perturbación del sueño, día y anochecer	30	16	-
Salas de tratamiento en hospitales, interior	Interferencia con descanso y restablecimiento	Tan débil como se pueda		

⁶Fuente: ArmstrongT:Cumulative trauma disorders of the hand and wrist.AIHA ergonomics guide.AIHA 1994

Zonas industriales, comerciales y de tráfico, interior y exterior	Daños al oído	70	24	110
Ceremonias, festivales y actividades recreativas	Daños al oído (asistentes habituales: < 5 veces/año)	100	4	110
Altavoces, interior y exterior	Daños al oído	85	1	110
Música a través de cascos y auriculares	Daños al oído (valores en campo libre)	85	1	110
		Bajo los cascos, adaptada a campo libre		
Sonidos impulsivos de juguetes, fuegos artificiales y armas de fuego	Daños al oído (adultos)	-	-	140
Exteriores en parques y áreas protegidas	Perturbación de la tranquilidad	Las zonas tranquilas exteriores deben preservarse y minimizar en ellas la razón de ruido perturbador a sonido natural de fondo		

CAPITULO II

2. SOFTWARE Y ENTORNOS DE DESARROLLO

2.1. Introducción

En la actualidad existe una gran cantidad de entornos de desarrollo, que facilitan en gran manera el diseño y desarrollo de software. La disminución de los tiempos que se emplean en el diseño, construcción e implementación, de los sistemas se ha convertido en uno de los principales aspectos de análisis a la hora de elegir el entorno de desarrollo. A más de la disminución de tiempos siempre se ha pensado en la escalabilidad y la proyección o tendencia del entorno de desarrollo, de hecho los únicos tipos de software entornos de desarrollo que se han logrado mantener en vanguardia, justamente han sido aquellos que han podido escalar y evolucionar a la par con las novedades tecnológicas de hardware.

Paralelamente al desarrollo tecnológico de hardware, está la parte de software que manejen y optimicen estos hardwares, es decir no se podría hablar de optimización de procesos, aislado el uno del otro, aquello implica que los entornos de desarrollo estén también preparados, para el desarrollo de software en varios dispositivos.

2.2. TECNOLOGÍAS PARA EL DESARROLLO DE APLICACIONES

Las tecnologías actuales de desarrollo se han visto obligadas a cumplir ciertos parámetros y normas de calidad, para poder mantenerse en el mundo competitivo de las TIC's mismas que están en constante evolución.

Dentro de los principales punto de análisis se encuentran: la arquitectura, la escalabilidad, la integridad, la disponibilidad.

2.2.1. Arquitectura de Software

En los inicios de la informática, la programación se consideraba un arte y se desarrollaba como tal, debido a la dificultad que entrañaba para la mayoría de las personas, pero con el tiempo se han ido descubriendo y desarrollando formas y guías generales, con base a las cuales se puedan resolver los problemas. A estas guías, se les ha denominado Arquitectura de Software, porque, si observamos la semejanza de los planos de un edificio o construcción, estas indican la estructura, funcionamiento e interacción entre las partes del software.

La arquitectura de lenguajes de programación⁷ comprende, el diseño y especificación de la estructura global del sistema, dentro de los cuales existen dos tipos de arquitecturas de lenguajes de programación: Los lenguajes de programación de bajo nivel y los de alto nivel.

2.2.1.1. Arquitectura de bajo nivel

El tipo de lenguaje de programación de bajo nivel depende totalmente de la máquina, la cual entiende el lenguaje binario o el código máquina, que consiste en ceros y unos. Es decir, que para realizar cualquier acción, solo utilizan ceros y unos. Los lenguajes de programación más cercanos a la arquitectura del hardware pertenecen al tipo de lenguajes de programación de bajo nivel.

El tipo de lenguaje de programación de bajo nivel es totalmente dependiente de la computadora, es decir que no podemos utilizarlo en cualquier otra. Este tipo de lenguaje de programación está prácticamente diseñado a la medida del hardware y aprovecha las características de este. Dentro de este tipo de lenguajes de programación podemos citar al lenguaje máquina y al lenguaje ensamblador.

El lenguaje utilizado por el procesador se denomina lenguaje máquina, el cual no es comprensible para los seres humanos, razón por la cual se han desarrollado lenguajes intermediarios comprensibles para el hombre. El código escrito en este tipo de lenguaje se transforma en código máquina para que el procesador pueda entenderlo y procesarlo.

El ensamblador fue el primer lenguaje de programación utilizado. Es muy similar al lenguaje máquina, pero los desarrolladores pueden comprenderlo. No obstante, este lenguaje se parece tanto al lenguaje máquina que depende estrictamente del tipo de procesador utilizado (cada tipo de procesador puede tener su propio lenguaje máquina). Así, un programa desarrollado para un equipo no puede ser *portado* a otro tipo de equipo.

⁷Un lenguaje de programación: es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutarse. Entonces, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo.

2.2.1.2.Arquitectura de alto nivel

Dentro del tipo de lenguajes de programación de alto nivel tenemos a todos aquellos lenguajes de programación que son más afines al lenguaje natural que al lenguaje máquina. Estos lenguajes de programación son completamente independientes de la arquitectura del hardware de la computadora. Por lo que en general, un programa escrito con un lenguaje de programación de alto nivel lo podemos utilizar en cualquier otra computadora.

Estos lenguajes de programación de alto nivel solo necesitan un traductor que entienda el código fuente y las características de la máquina para poder funcionar correctamente.

Los lenguajes que usan los equipos para comunicarse entre ellos no tienen nada que ver con los lenguajes de programación; se los conoce como protocolos de comunicación. Se trata de dos conceptos totalmente diferentes. Un lenguaje de programación es muy estricto:

Debido a que para cada instrucción le corresponde una acción de procesador. Por lo tanto, un lenguaje de programación tiene varias ventajas:

- Es más fácil de comprender que un lenguaje máquina:
- Permite mayor portabilidad, es decir que puede adaptarse fácilmente para ejecutarse en diferentes tipos de equipos.

2.2.1.2.1. Lenguajes de programación de alto nivel

Los lenguajes de programación generalmente se dividen en dos grupos principales en base al procesamiento de sus comandos:

- lenguajes imperativos;
- lenguajes funcionales.

2.2.1.2.1.1.Lenguaje de programación imperativo

Un lenguaje imperativo programa mediante una serie de comandos, agrupados en bloques y compuestos de órdenes condicionales que permiten al programa retornar a un bloque de

comandos si se cumple la condición. Estos fueron los primeros lenguajes de programación en uso y aún hoy muchos lenguajes modernos usan este principio.

No obstante, los lenguajes imperativos estructurados carecen de flexibilidad debido a la secuencialidad de las instrucciones.

2.2.1.2.1.2. Lenguaje de programación funcional

Lenguaje de programación funcional o lenguaje procedimental es un lenguaje que crea programas mediante funciones, devuelve un nuevo estado de resultado y recibe como entrada el resultado de otras funciones. Cuando una función se invoca a sí misma, hablamos de recursividad.

Tabla 3 Algunos ejemplos de lenguajes ampliamente usados⁸

Lenguaje	Principal área de aplicación	Compilado/interpretado
ADA	Tiempo real	Lenguaje compilado
BASIC	Programación para fines educativos	Lenguaje interpretado
C	Programación de sistema	Lenguaje compilado
C++	Programación de sistema orientado a objeto	Lenguaje compilado
Cobol	Administración	Lenguaje compilado
Fortran	Cálculo	Lenguaje compilado
Java	Programación orientada a Internet	Lenguaje intermediario
MATLAB	Cálculos matemáticos	Lenguaje interpretado
Cálcs matemáticos	Cálculos matemáticos	Lenguaje interpretado
LISP	Inteligencia artificial	Lenguaje intermediario
Pascal	Educación	Lenguaje compilado
PHP	Desarrollo de sitios web dinámicos	Lenguaje interpretado
IA	Inteligencia artificial	Lenguaje interpretado
Perl	Procesamiento de cadenas de caracteres	Lenguaje interpretado

2.2.2. Escalabilidad

Al igual que al obtener un sistema empresarial, o algún software de uso personal, necesitamos conocer su vigencia y tendencia, es decir si aquel software tiene una estabilidad y si constantemente está evolucionando, aquello se llama escalabilidad, esto mismo se toma en cuenta al momento de

⁸ Lenguajes de programación de Kioskea.net (es.kioskea.net) actualización 16 de octubre de 2008 bajo Licencia CreativeCommons de Kioskea.net Contrato V2.0 Atribución - No comercial - ShareAlike 2.0

escoger nuestro entorno de desarrollo, Debido al auge de las tecnologías y a la gran cantidad de entornos de desarrollo, que actualmente existe, se ha convertido la escalabilidad en un tema fundamental de análisis, el alcance y visión a posteriori son unos de los principios básicos de desarrollo actual, los grandes equipos de desarrollo que han logrado tener su madurez a lo largo de su experiencia saben que el principal enfoque y objetividad de desarrollo está orientado a dar un mantenimiento rápido y eficaz a más de la posibilidad de poder evolucionar, y ser mejorado.

2.2.3. Integridad

Actualmente la mayoría de los entornos de desarrollo traen consigo un sistema inteligente a parte del compilador, propio de los entornos de desarrollo, estos sistemas inteligente conoce la sintaxis y la semántica del lenguaje de programación en uso, y tienen la capacidad de predecir y auto completar frases o palabras acorde a lo que se esté escribiendo. Estos sistemas inteligentes nos garantizan la integridad de las aplicaciones que desarrollemos ya que todo el proceso e interpretación de sentencias y conversión en el lenguaje máquina es realizado por el entorno de desarrollo y el lenguaje de programación del que se está haciendo uso. Hoy en día existen muchos entornos de desarrollo que soportan no solo uno si no varios lenguajes de programación.

2.2.4. Disponibilidad

Es quizás el aspecto más importante al momento de elegir nuestro entorno de desarrollo, ya que así tengamos todas las características anteriormente descritas en un entorno de desarrollo, esto no nos sirve de nada si aquella herramienta no está disponible o accesible a nosotros, coincidentalmente las herramientas o entornos de desarrollo que más nos facilitan el trabajo de desarrollo son a su vez menos accesible ya sea por costos o por comprensión de y uso de la herramienta o limitaciones de hardware. No obstante con visión en capturar la mayor cantidad de usuarios de sus herramientas la mayoría de los creadores de entornos de desarrollo propietarios han sacado sus versiones express (gratuitas), dependiendo de la aplicación a desarrollar estas versiones tratan de cumplir con los requerimientos necesarios anteriormente descritos.

2.3. ELECCION DE LA TECNOLOGIA

2.3.1. Open Source Vs Software⁹Propietario

2.3.1.1. Open source

Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre¹⁰.

2.3.1.1.1. Ventajas del software libre.

- Existen aplicaciones para todas las plataformas (Linux, Windows, Mac).
- El precio de las aplicaciones es mucho menor, la mayoría de las veces son gratuitas.
- Libertad de copia.
- Libertad de modificación y mejora.
- Libertad de uso con cualquier fin.
- Libertad de redistribución.
- Facilidad a la hora de traducir una aplicación en varios idiomas.
- Mayor seguridad y fiabilidad.
- El usuario no depende del autor del software.

2.3.1.1.2. Desventajas del software libre

- Algunas aplicaciones (bajo Linux) pueden llegar a ser algo complicadas de instalar.
- Inexistencia de garantía por parte del autor.
- Interfaces gráficas menos amigables.
- Poca estabilidad y flexibilidad en el campo de multimedia.
- Menor compatibilidad con el hardware.

⁹Software es la suma total de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de cómputo.

¹⁰Su uso nació por primera vez en 1998 de la mano de algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (*free software*). *Free* en inglés significa dos cosas distintas dependiendo del contexto: gratuidad y libertad. Lo cual implica, para el caso que nos ocupa, "software que podemos leer, modificar y redistribuir gratuitamente"

2.3.1.2. Software propietario

El software no libre también es llamado software propietario, software privativo, software privado o software con propietario. Se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o que su código fuente no está disponible o el acceso a este se encuentra restringido.

2.3.1.2.1. Ventajas del software propietario.

- Facilidad de adquisición (puede venir preinstalado con la compra del computador, o encontrarlo fácilmente en las tiendas)
- Existencia de programas diseñados específicamente para desarrollar una tarea.
- Las empresas que desarrollan este tipo de software son por lo general grandes y pueden dedicar muchos recursos, sobretodo económicos, en el desarrollo e investigación.
- Interfaces gráficas mejor diseñadas.
- Más compatibilidad en el terreno de multimedia y juegos.
- Mayor compatibilidad con el hardware.

2.3.1.2.2. Desventajas del software propietario

- No existen aplicaciones para todas las plataformas (Linux y Mac OS).
- Imposibilidad de copia.
- Imposibilidad de modificación.
- Restricciones en el uso (marcadas por la licencia).
- Imposibilidad de redistribución.
- Por lo general suelen ser menos seguras.
- El coste de las aplicaciones es mayor.
- El soporte de la aplicación es exclusivo del propietario.
- El usuario que adquiere software propietario depende al 100% de la empresa propietaria.

2.3.2. Entornos de desarrollo (IDE's)¹¹

Un entorno de desarrollo integrado es un programa de cómputo compuesto por un conjunto de herramientas de programación.

¹¹IDE sigla en inglés de integrateddevelopmentenvironment

Los IDE's pueden dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios lenguajes. Actualmente los IDE's buscan ser aptos para distintos lenguajes de programación, es común que en un entorno de desarrollo integrado se encuentre compatibilidad y soporte para más de un lenguaje programación, no obstante existen herramientas completas dedicadas exclusivamente a un lenguaje, del cual quieren brindar las mayores facilidades para el desarrollo.

2.3.3. Comparación funcional

Microsoft Visual Studio es un IDE para sistemas operativos Windows. Que soporta varios lenguajes de programación tales como: Visual C++, Visual C#, Visual J#, ASP.NET, y Visual Basic.NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión net 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

Actualmente existen varios IDE's que soportan estos lenguajes, por ejemplo para el lenguaje de programación C# existen varios proyectos de distribución libre, algunos de esos proyectos son: Sharp Developer, el ya conocido proyecto Mono, que si bien es cierto trae soporte para sistemas operativos GNU/Linux, aún no ha alcanzado la madurez necesaria para competir con Visual Studio.

Dentro de los principales problemas, que tiene estos proyectos es el manejo de Ventanas, al ser Visual Studio una herramienta de Microsoft se complementa perfectamente el manejo de ventanas con el sistema operativo Windows de la Microsoft, mientras que el proyecto Mono, contiene el gestos de modo grafico GTK¹², que se obliga a funcionar exclusivamente en la plataforma de desarrollo y si deseamos moverlo a otra plataforma, tendríamos que recompilar y modificar nuestro aspecto gráfico, lo cual implica un tiempo adicional en el desarrollo.

La facilidad manejo y desarrollo que posee Visual Studio, disminuye evidentemente el tiempo empleado en el desarrollo, ya que sus

¹²GTK "GIMP Tool Kit" es una biblioteca del equipo GTK+, la cual contiene los objetos y funciones para crear la interfaz gráfica de usuario. Manejajwidgets(pequeña aplicación o programa) como ventanas, botones, menús, etiquetas, deslizadores, pestañas, etc.

herramientas, están orientadas más al diseño, en vista de que sus sistemas inteligentes generan, gran cantidad de código, compitiendo así con una herramienta CASE¹³, pero con el valor agregado de la flexibilidad y completa personalización de nuestros programas.

2.4. APRENDIZAJE DE LA TECNOLOGÍA ESCOGIDA

2.4.1. Componentes del entorno de desarrollo

Pantalla inicial: en la pantalla inicial nos muestra una interfaz con las aplicaciones recientes, y con posibilidad de ignorar esto para crear nuevas aplicaciones.

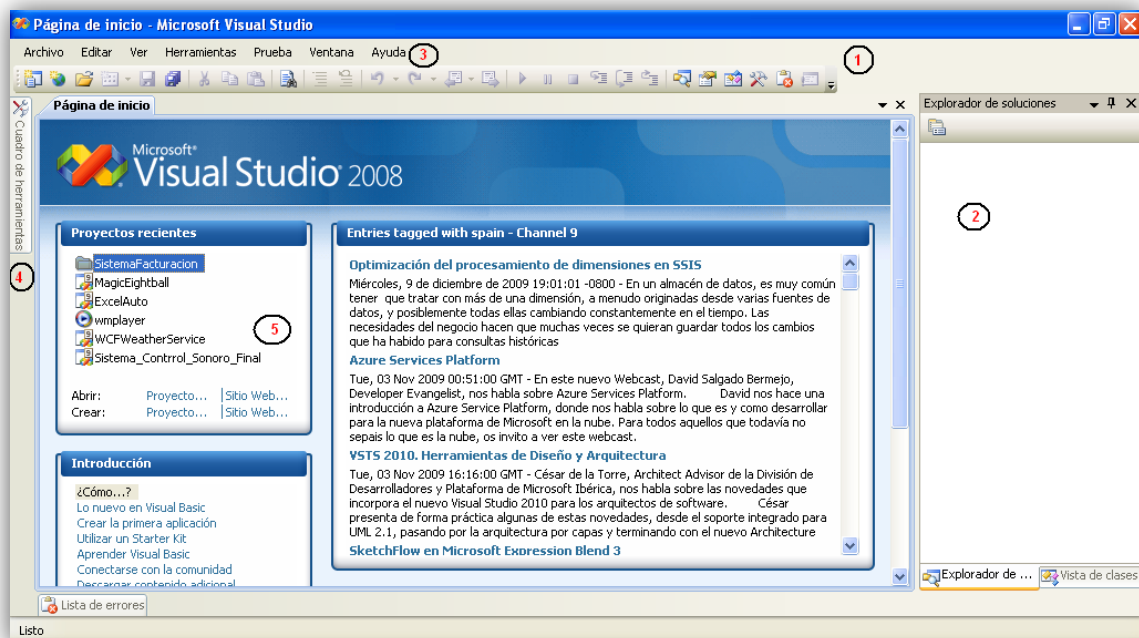


Figura 2 Pantalla Principal del entorno de desarrollo MS Visual Studio .NET 2008

En este primer vistazo del entorno de desarrollo de Microsoft (Visual Studio), mostramos a continuación las principales partes que en este existen.

- Barra de Herramientas: en esta barra encontramos los principales iconos de las más utilizadas tareas y actividades que nos sirven dentro del desarrollo de nuestro software.

¹³Las herramientas CASE (*ComputerAidedSoftware Engineering*, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

- Explorador de soluciones: como el entorno Visual Studio soporta varios lenguajes de programación podemos tener varias soluciones, en diferentes lenguajes, en esta porción de la ventana nos muestran las diferentes soluciones a más de las propiedades de los objetos que se están usando en nuestro proyecto.
- La Barra de Menús: nos muestran los Menús principales del entorno de desarrollo, aquí se muestran las opciones de desarrollo, fundamentales, para el desarrollo rápido de una manera, eficaz y dinámicamente.
- Cuadro de Herramientas: aquí se muestran los controles que podemos agregar a nuestro proyecto, dependiendo del tipo de aplicación que estemos desarrollando, es decir si se trata de aplicaciones de escritorio, o aplicaciones Web.
- Proyectos recientes: nos muestra los proyectos que han sido abiertos más recientemente, con posibilidad también de poder crear, y/o abrir nuevos proyectos.

2.4.2. Aplicación de ejemplo

Lo primero que vamos a hacer es cargar el entorno de desarrollo del Visual Studio.NET, así que vamos a Inicio - Programas - Microsoft Visual Basic 2008... y lo ejecutamos. En la pantalla de inicio volvemos a seleccionar "Aplicación de Windows Forms", como se ve en la imagen.

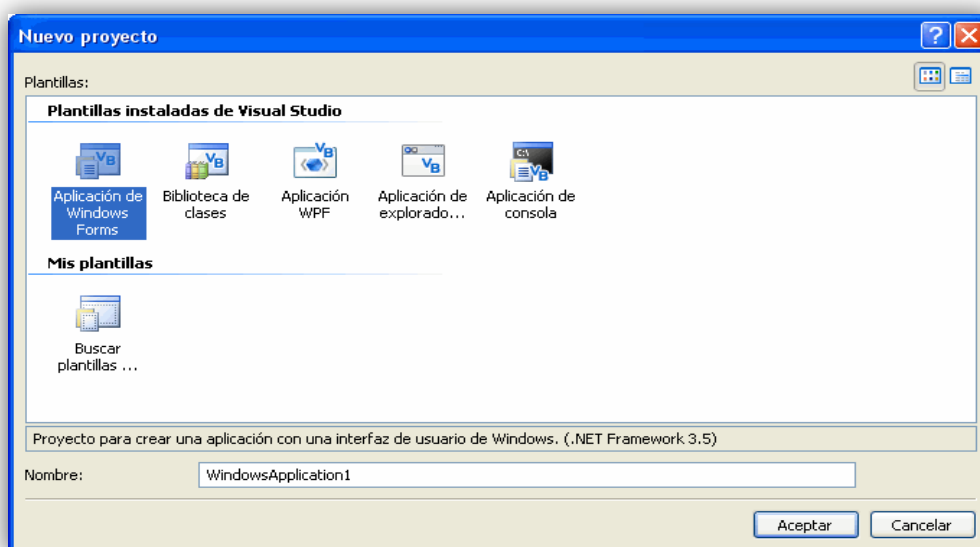


Figura 3 Crear una aplicación de ejemplo

En el proyecto, se habrá cargado un formulario que se te mostrará de forma automática.

En el lado derecho de la pantalla, está el panel de Explorador de soluciones en la que se indica el proyecto actual y se muestran los ficheros que lo componen. Ese panel es el Explorador de Soluciones.

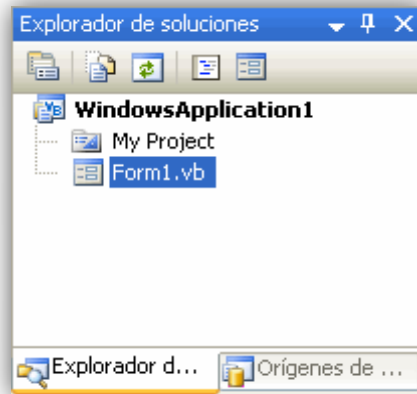


Figura 4 Panel Explorador de proyectos MS Visual Studio

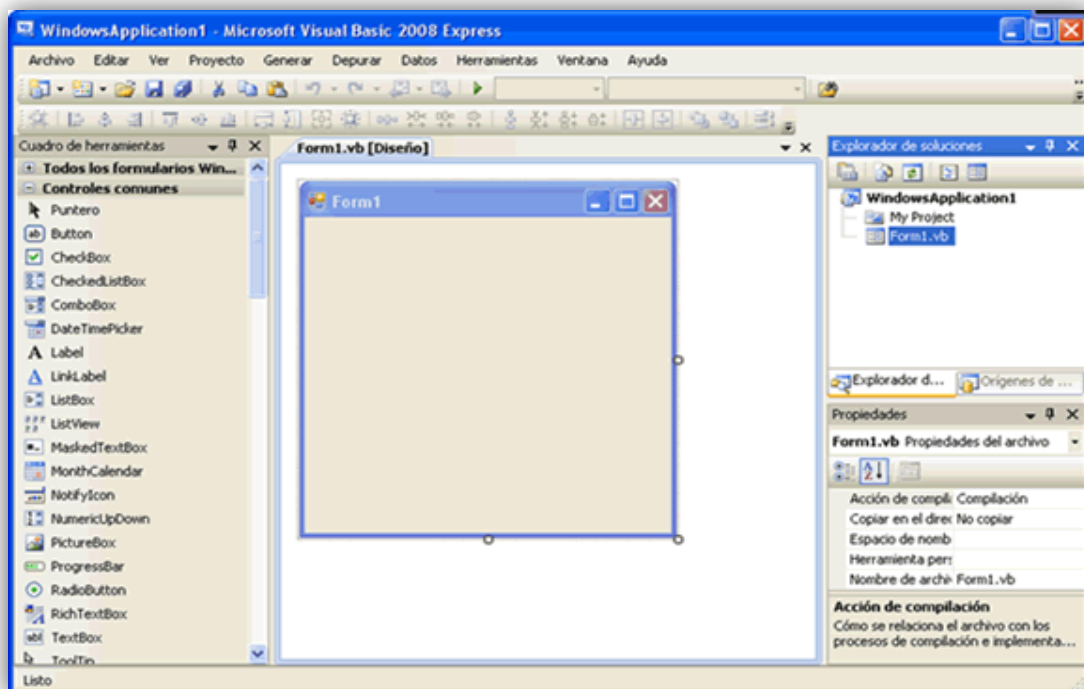


Figura 5 Pantalla completa Proyecto de prueba

Donde distinguimos en la parte izquierda el cuadro de herramientas, en el centro el formulario principal y en la parte derecha las ventanas de propiedades y el explorador de soluciones.

Los controles: son los elementos que insertamos dentro de un formulario y que nos va a permitir interactuar entre el usuario y el código: botones, cuadros de texto, etiquetas, cuadros desplegables, cuadrículas de datos. En definitiva todos y cada uno de los elementos que vemos en los formularios de todas las aplicaciones. La lista de controles básicos disponibles la tenemos a la izquierda, en el panel de herramientas.

2.4.3. Empezando la aplicación

Para añadir controles al formulario utilizaremos la barra de herramientas. Por ejemplo, para añadir una etiqueta (Label) y una caja de texto (TextBox), simplemente damos doble-clic sobre esos elementos de la barra de herramientas y se añadirán al formulario o podemos también agarrarlos y arrastrarlos hasta nuestro formulario.

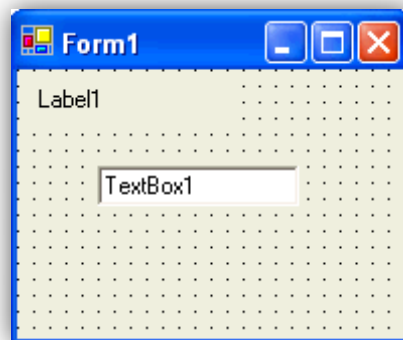


Figura 6 Formulario del proyecto de prueba Campo de texto

Para nuestro ejemplo, añadiremos un botón (Button) y lo situaremos debajo del cuadro de texto (Textbox). Luego añadimos una etiqueta (Label) para que quede de esta forma:

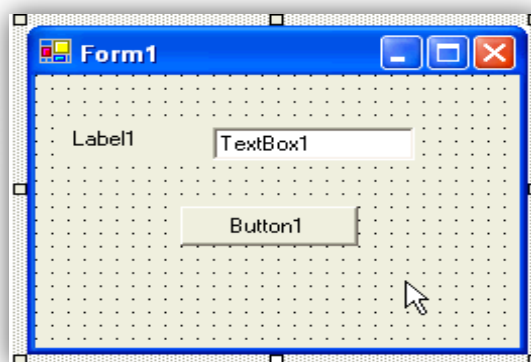


Figura 7 Formulario del proyecto de prueba Botón

Por defecto, el IDE asigna unos nombres genéricos a los controles, como "label1", "textbox1", "button1", es decir utiliza el tipo de control y lo va numerando tantas veces como controles iguales exista en el formulario (label1, label2, label3,...). Pero estos nombres al igual que las demás propiedades del control son configurables. Para el uso de los nombre se recomienda que el control tenga un nombre descriptivo, correspondiente a la función que realiza en la aplicación.

Ahora vamos a cambiar el texto que contiene el botón "Button1". Para cambiarle este texto hay que utilizar la ventana de propiedades (A la derecha-abajo), en esta ocasión el elemento que nos interesa de esa ventana de propiedades es Text, escribimos en esta propiedad la palabra "Mostrar" y cuando pulses Intro(Enter) o el tabulador veremos que el texto del botón se ha actualizado.

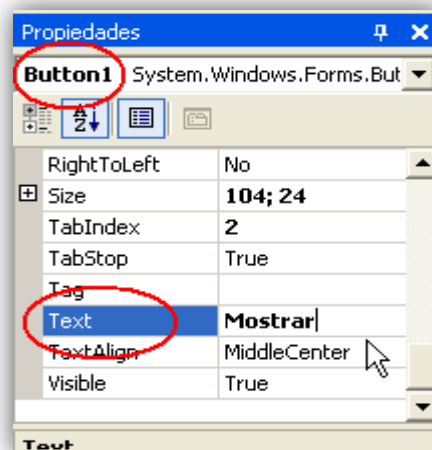


Figura 8 Panel de Propiedades del campo del botón

Hacemos lo mismo con la etiqueta "Label", hay que seleccionarla primero haciendo clic para que se muestren las propiedades de la etiqueta. Escribimos "Nombre:" en Text y pulsamos intro(enter) o el tabulador.

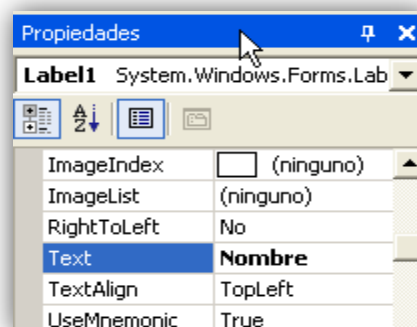


Figura 9 Panel de Propiedades del Label

En la parte superior de la ventana se muestra, el tipo de control que se está editando y su nombre.

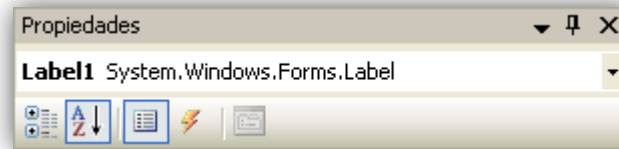


Figura 10 Parte Superior del Panel de propiedades

2.4.4. Escribir el código

Ahora vamos a escribir el código que se ejecutará cada vez que se haga clic en el botón que hemos añadido. Para ello, selecciona el botón "Mostrar" y hacemos doble clic en él, se mostrará una nueva ventana, en este caso la ventana de código del botón, asociada con el formulario que tenemos en nuestro proyecto. Nos mostrará:

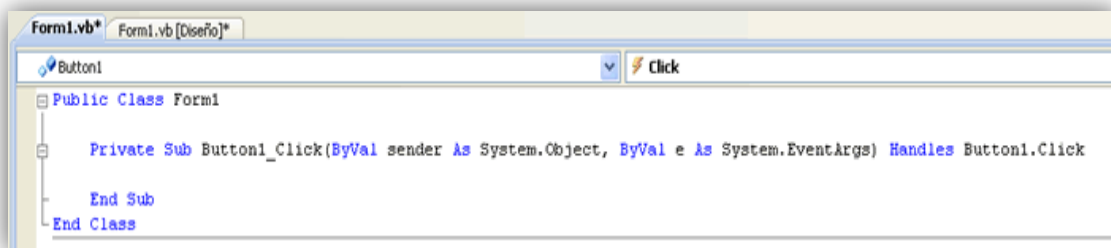


Figura 11 Porción de Código (Button)

Cuando se da doble clic sobre el control que se haya seleccionado Visual Studio entiende que la solicitud es una petición para mostrar la porción de código dentro de la aplicación o solución. También se puede acceder a la ventana de código presionado clic con el botón secundario sobre el control que se desee.

El panel de propiedades no solo nos permite modificar la apariencia y características del control sino que también nos permite escoger los eventos que se pueden generar sobre el control seleccionado.

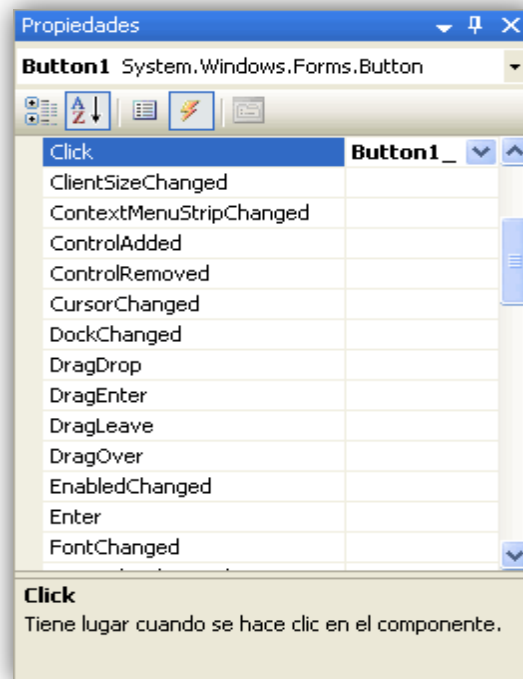


Figura 12 Propiedad de Eventos del Botón

Otra forma de ingresar a la ventana de código es dentro del panel de explorador de soluciones existen varios botones uno de ellos es el “Ver código”

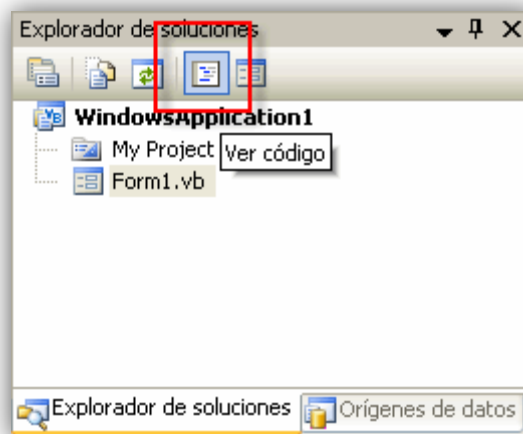


Figura 13 Explorador de soluciones Ver código

En la parte izquierda de la ventana del código que se puede escoger el control sobre el cual se desee ingresar el código y en la parte derecha de la ventana nos despliega los posibles eventos sobre los cuales

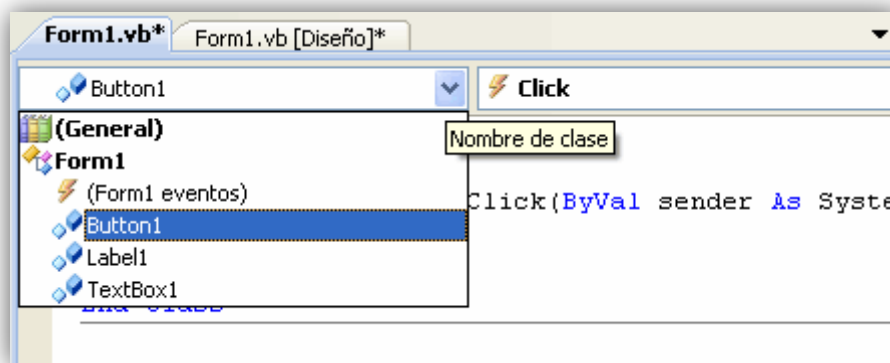


Figura 14 Navegador de Objetos dentro del código

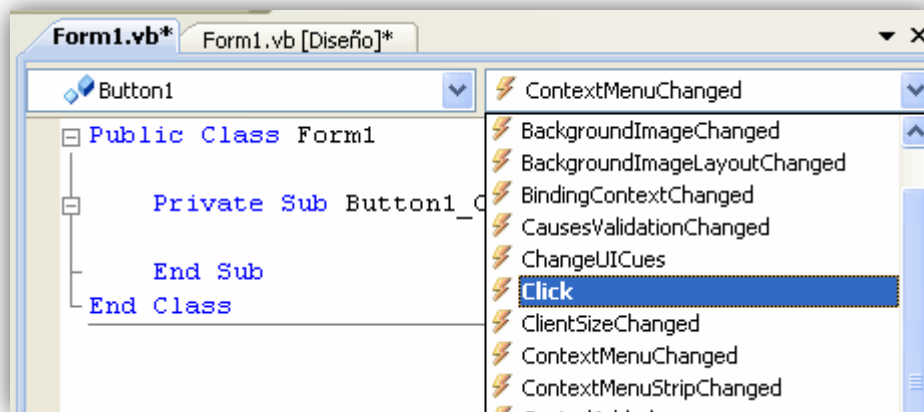


Figura 15 Listado de eventos del objeto seleccionado

Volvemos a la pantalla del código para el evento clic del botón

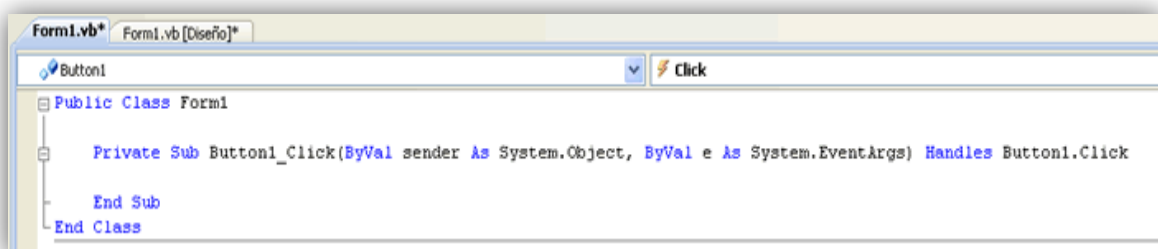


Figura 16 Código fuente botón

Esta pantalla muestra en la parte superior, que el objeto o clase de referencia "Form1.vb". Un poco más abajo se muestra el control de referencia (a lado izquierdo) llamado "Button1" y a la derecha que estamos trabajando con el evento click.

Escribimos el siguiente código en el espacio delimitado por "Private Sub... y End Sub"

```
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object,
        MsgBox("Hola: " & TextBox1.Text)
    End Sub
End Class
```

Figura 17 Porción de código de prueba

Pulsamos F5 para que se ejecute el código que hemos escrito o pulsa en el botón "play" que está en la barra de botones.

Cuando se presente el formulario se ingresa el texto "Jose", pulsa en el botón Mostrar. Entonces se muestra un cuadro de diálogo con el texto "Hola" seguido del texto ingresado en el campo de texto (TextBox).

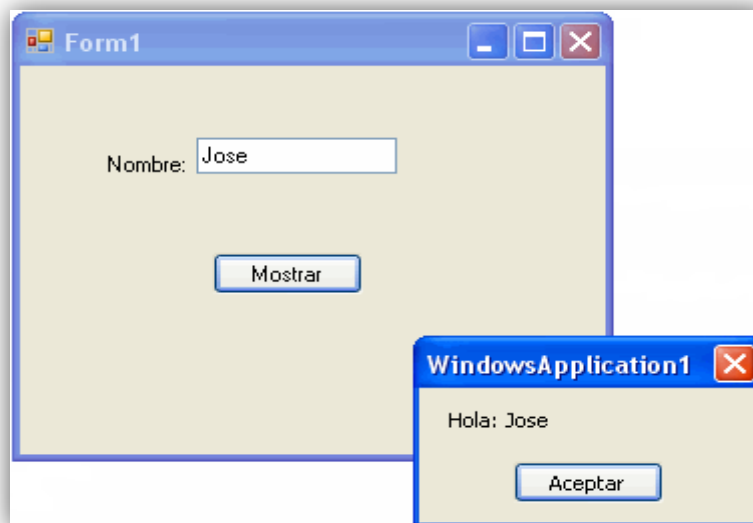


Figura 18 Programa de Prueba en tiempo de ejecución

CAPITULO 3

3. ESTRUCTURA Y DISEÑO DEL SISTEMA DE CONTROL SONORO

3.1. Introducción

En el capítulo anterior hemos tratado las diferentes formas de representar a los sistemas, con lo cual estamos listos para mostrar de forma estructurada y detallada al sistema de control sonoro. Lo que se pretende es documentar cada una de las instancias del sistema, de manera que en lo posterior pueda ser implementado, modificado, y/o expandido. Cumpliendo con las reglas y normas propias del lenguaje U.M.L detallado anteriormente, se muestra el sistema de control sonoro en cada una de las etapas. En primera instancia se detalla el ámbito del proyecto, así como los requerimientos globales del sistema.

3.2. AMBITO DEL PROYECTO:

El sistema de control sonoro está dividido en dos partes: hardware y software, que funcionan aisladamente pero que necesariamente se unen para obtener el tipo de información requerida, por lo cual se detalla cada una de las dos partes:

3.2.1. Hardware

El sistema consta de un dispositivo electrónico, instalado en cada una de las unidades de transporte (UDT), el cual registra información acerca del uso de la bocina, el dispositivo consta de una pequeña memoria eprom que almacena la información temporalmente hasta que esta sea descargada, mediante el software del sistema. Además contiene un reloj de tiempo real, que me permite registrar cada suceso, en el que intervenga la bocina, esto es la intensidad en dB de sonido, la duración, el número de veces, etc., además el dispositivo permite bloquear la bocina si el conductor ha excedido en el uso de la misma, aunque también puede ser bloqueado y desbloqueado manualmente. Este dispositivo es alimentado con la misma batería de la unidad de transporte por lo que no se requiere de energía externa, la placa del dispositivo consta de un puerto serial db9¹⁴ que me permite establecer comunicación con el computador y descargarme en este la información.

El dispositivo contiene información totalmente configurable, como clave de acceso, la fecha actual, las opciones, la cantidad de veces que el dispositivo necesita para bloquearse etc.

A continuación se detalla los elementos que conforman el dispositivo electrónico así como el esquema del circuito que este involucra:

¹⁴El conector **DB9** es un conector analógico de 9 clavijas .

Tabla 4 Elementos electrónicos del dispositivo

Cant.	Descripción
9	Capacitor cerámico de 0.1uF (microFaradios)
2	Capacitor cerámico de 68nF (nanoFaradios)
2	Capacitor cerámico de 22pF (picoFaradios)
6	Capacitor electrolítico de 10uF / 25V
1	Capacitor electrolítico de 2.2uF / 25V
1	Capacitor electrolítico de 220uF / 25V
1	Resistencia de 100 óhmios / 0.25W
2	Resistencia de 10 kilo-óhmios / 0.25W
1	Resistencia de 12 kilo-óhmios / 0.25W
1	Resistencia de 2.4 kilo-óhmios / 0.25W
1	Resistencia de 2.7 kilo-óhmios / 0.25W
4	Resistencia de 390 óhmios / 0.25W
11	Resistencia de 4.7 kilo-óhmios / 0.25W
1	Resistencia de 47 óhmios / 2W
2	Resistencia de 5.6 kilo-óhmios / 0.25W
1	Resistencia de 56 óhmios / 0.25W
1	Resistencia de 6.8 kilo-óhmios / 0.25W
1	Potenciómetro de precisión pequeño de 100 kilo-óhmios
2	Potenciómetro de precisión pequeño de 20 kilo-óhmios
2	Diodo rectificador 1N4007
2	Diodo rápido 1N4148
1	Diodo Led de 1.5V / 20mA Rojo
1	Diodo Led de 1.5V / 20mA Amarillo
1	Diodo Led de 1.5V / 20mA Verde
1	Diodo Led de 1.5V / 20mA Azul
8	Jumper
1	Peineta para jumper de 40 pines
1	Cristal de Cuarzo de 20MHz
1	Cristal de Cuarzo de 32.768KHz
5	Pulsantes de dos pines blancos redondos
2	C.I. Memoria EEPROM 24LC256
1	C.I. Regulador de Voltaje 7805
1	C.I. RTC (Reloj de Tiempo Real) DS1307
1	C.I. Pic 16F876A
2	C.I. Amplificador Operacional LM358
1	C.I. Conversor TTL//RS232 MAX232
1	C.I. Optotransistor PS2501-2
1	Relé de 12Vdc con salida a 125Vac / 10A
1	LCD con backlight (rojo ó azul ó amarillo)
2	Transistor 2N3904
1	Batería de Litio de 3V CR2016 con su respectiva porta batería

3.2.2. Características físicas de dispositivo:

- a) Dimensiones $x=15\text{ cm}$ $y=10\text{ cm}$ $z=2\text{ cm}$
- b) Contiene tornillos de soporte
- c) Pantalla digital que visualiza los procesos

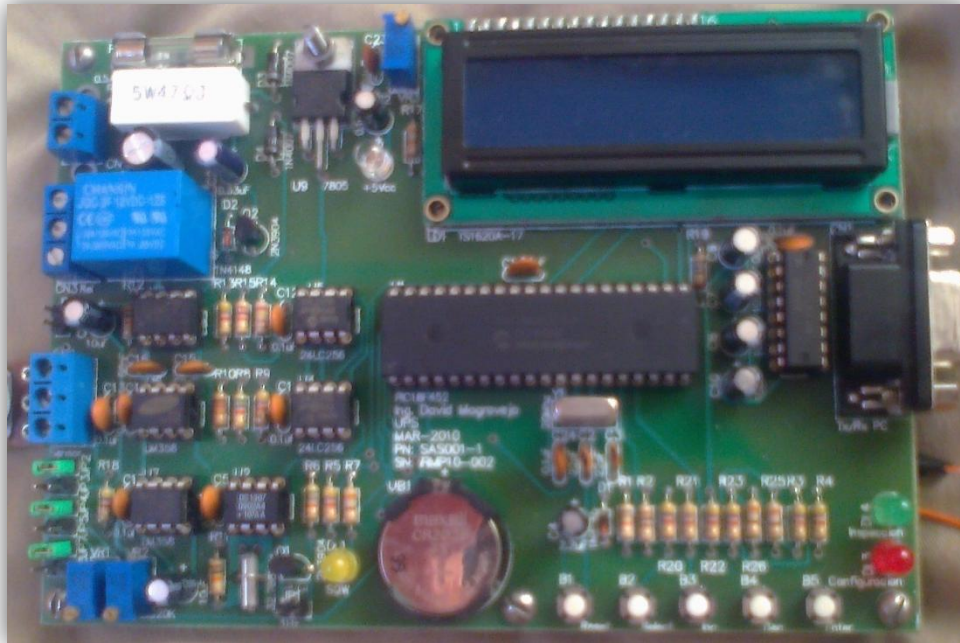


Figura 19 Vista aérea del dispositivo electrónico

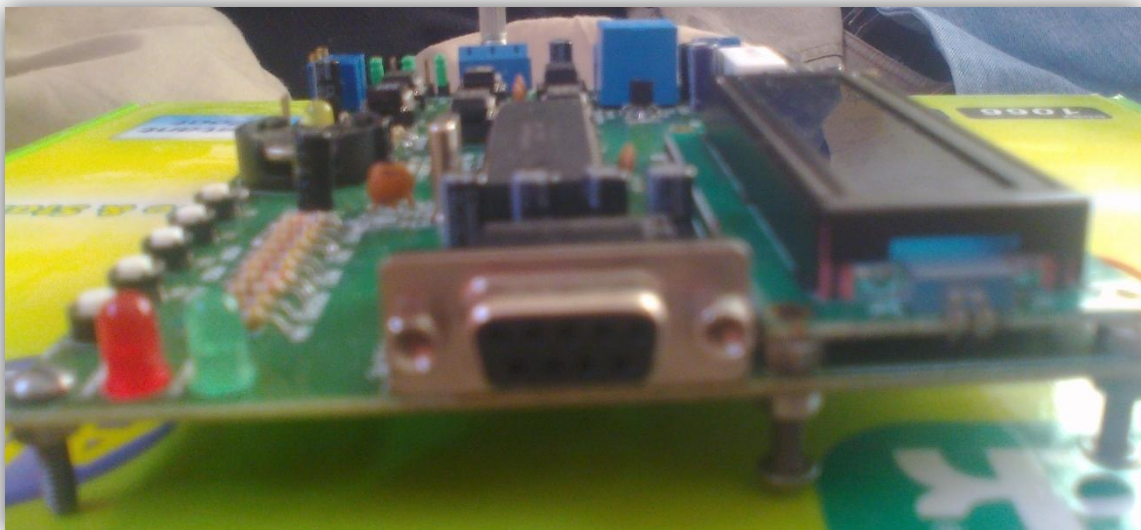


Figura 20 Perspectiva frontal del dispositivo electrónico

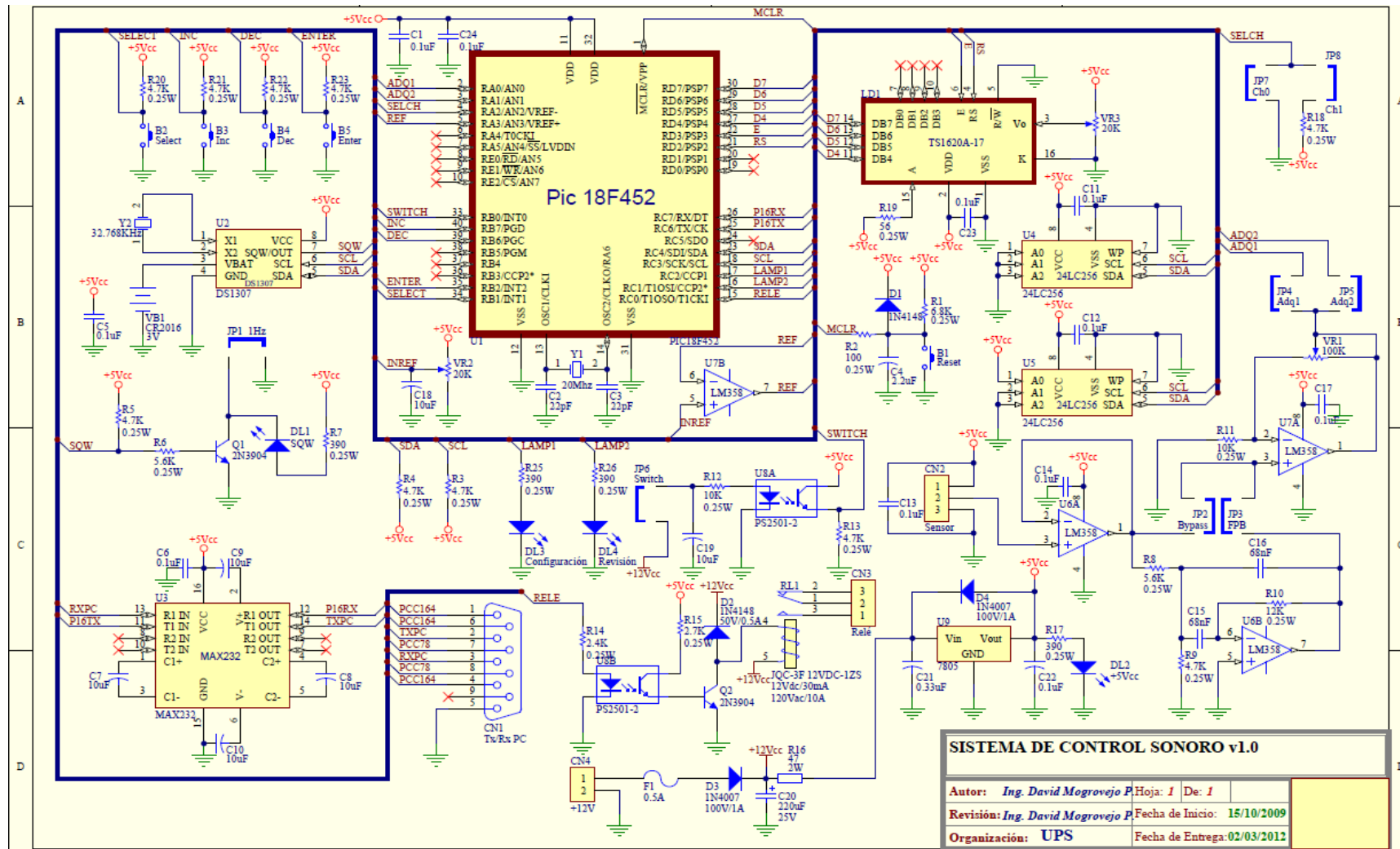


Figura 21 Esquema del Circuito eléctrico del dispositivo de adquisición de sonido

3.2.3. Software

El sistema permite interactuar con el hardware a más de administrar y tomar decisiones en base a los resultados obtenidos del dispositivo. Consta de dos módulos esenciales uno que se encarga de la parte netamente administrativa, y la otra que nos permite realizar análisis sobre la información.

En el primer caso el sistema me ayuda a almacenar información sobre las cooperativas de transportes, choferes, rutas y recorridos de los buses. Así como la creación y administración de usuarios del sistema.

En el segundo caso tenemos el módulo de uso de bocinas que contiene herramientas para generar reportes y gráficos estadísticos, así como herramientas que permiten migrar la información desde el dispositivo a la base de datos.

El sistema contiene las siguientes funcionalidades:

- gestión de cooperativas
- gestión de representantes
- gestión de rutas
- gestión de buses
- gestión de uso bocinas
- gestión de choferes
- gestión de usuarios

3.2.4. GESTION DE COOPERATIVAS:

Maneja información sobre las cooperativas de transporte público de la ciudad de Cuenca, es el eje principal de la gestión del sistema ya que esta gestión engloba gran parte de información esencial del sistema.

3.2.4.1. Información básica del grupo:

PROPIA

- ID cooperativas: identificador único del sistema
- Ruc de la cooperativa: como empresa o entidad legal ante el SRI, cada organización o entidad empresarial necesita un identificador en este caso es el RUC

- Nombre de la cooperativa: conocido también como razón social a más del identificador la entidad necesita un nombre que lo identifique ante otras entidades.
- Observación de la cooperativa: son detalles y observaciones que se pueden dar al momento de crear, editar o eliminar una cooperativa.

EXTERNA

- Direcciones:
 - Id de Dirección: identificador único del sistema
 - Calle Principal: principal descripción de la dirección
 - Calle Secundaria: descripción complementaria para una dirección exacta.
 - Numero Predio: identificación municipal.
- Teléfonos:
 - Id Teléfono: identificador único del sistema
 - Tipo Numero Teléfono: indica si el número es móvil o fijo(convencional)
 - Numero Teléfono: número necesario para el contacto, otorgado por la empresa proveedora de servicios telefónicos.
 - ID Cooperativa: identificador de la cooperativa al cual pertenece el número telefónico.
- Direcciones:
 - Id Email: identificador único del sistema
 - Dirección de Email: indica la dirección de correo electrónico.
 - Tipo de Cuenta: indica si la dirección de correo pertenece a una red social pública o privada.
 - ID Cooperativa: identificación que indica la cooperativa ala la cual pertenece.

3.2.4.2.Mantenimiento de cooperativas:

- Crear cooperativa
- Editar cooperativa
- Eliminar cooperativa
- Buscar cooperativa
- Mantenimiento de direcciones de cooperativas

- Crear dirección
- Editar dirección
- Eliminar dirección
- Mantenimiento de teléfonos de cooperativas
 - Crear teléfono
 - Editar teléfono
 - Eliminar teléfono
- Mantenimiento de direcciones de correo electrónico de las cooperativas
 - Crear email
 - Editar email
 - Eliminar email

3.2.5. GESTION DE BUSES:

Conocidas como unidades de transporte, son los medios de transporte en el cual se instalan los dispositivos de control sonoro.

3.2.5.1. Información básica de los buses:

- ID del bus: identificador único de las UDTs.¹⁵
- Marca del bus: indica la marca del vehículo.
- Modelo del bus: indica el año de fabricación de la UDT
- Placa del bus: indica el número de matriculo de la UDT
- Número del bus: cada UDT, perteneciente a alguna cooperativa tiene un número.
- ID de cooperativa: indica el id de la cooperativa al cual pertenece.

3.2.5.2. Mantenimiento de buses:

- Crear bus
- Editar bus
- Eliminar bus
- Buscar bus

3.2.6. GESTION CHOFERES

Es necesario saberla información del chofer que conduce la UDT.

¹⁵UDT: unidad de transporte, nombre de referencia de los buses o unidades de transporte.

3.2.6.1. Información básica del chofer:

PROPIA

- ID chofer: identificador único del sistema
- Cedula chofer: número de cedula del chofer
- Nombre del chofer: nombre completo del chofer
- Dirección del chofer: dirección domiciliaria del chofer
- Telefono1 del chofer: número telefónico del chofer (tipo móvil), ya que ahí recibirá mensajes de texto sobre el uso de la bocina que el chofer le ha dado a su unidad de transporte.
- Telefono2 del chofer: número telefónico alternativo, puede ser de tipo fijo o convencional
- Email del chofer: en caso de que el chofer posea una dirección de correo electrónico.
- Id del bus: Identificador del bus al cual el chofer está asociado.

EXTERNA

- ID licencia: identificador único del sistema.
- Tipo licencia: Describe el tipo de licencia que el chofer posee
- Tipo sangre: dato necesario de la licencia, pensado por los accidentes de tránsito o algún tipo de emergencia
- Fecha de emisión de la licencia: indica la fecha en la cual se obtuvo la licencia
- Fecha de caducidad de la licencia: indica el tiempo de vigencia en la cual tendrá validez la licencia.
- ID del chofer: indica el id del chofer al cual pertenece la licencia

3.2.6.2. Mantenimiento de chofer:

- Crear chofer
- Editar chofer
- Eliminar chofer
- Mantenimiento chofer
 - Crear licencia
 - Editar licencia
 - Eliminar licencia

3.2.7. GESTION DE USO DE BOCINAS

Cada UDT tiene instalado el dispositivo que registra el uso de las bocinas por lo cual necesitamos una gestión sobre el uso de bocinas.

3.2.7.1. Información básica del uso de la bocina:

- ID uso de bocina: identificador único dentro del sistema
- Hora uso de bocina: registra la hora, con minutos y segundos en los cuales se registró el uso de la bocina.
- Día de la semana: indica el día de la semana en el que se ha registrado el uso de la bocina.
- Fecha del uso de bocina: indica la fecha calendario en la cual se registró el uso de la bocina.
- Decibel alto del uso de la bocina: almacena el pico más alto que alcanzo al usar la bocina.
- Decibel bajo del uso de la bocina: obtiene un promedio entre el pico más alto y el más bajo.
- Decibel medio del uso de la bocina: almacena el pico más bajo en el cual se registró el sonido producido por el uso de la bocina.
- Duración milisegundos del uso de la bocina: indica el tiempo de duración que permaneció presionada la bocina, esto lo hace en milisegundos
- ID bus: Identificador del bus al cual pertenece el dispositivo
- ID dispositivo: identificador del dispositivo único en el sistema.
- ID usuario: almacena el usuario que realizo la transmisión de datos.

3.2.7.2. Mantenimiento de uso de bocina:

- Crear uso de bocina
- Consultas y reportes de uso de bocina

3.2.8. GESTION DE DISPOSITIVOS

Cada dispositivo mantiene información única por lo cual necesitamos una gestión de dispositivos.

3.2.8.1. Información básica de los dispositivos:

- ID dispositivo: identificador único en el sistema

- Serial del dispositivo: cada dispositivo sería fabricado con un número de serie.
- Fecha ingreso: la fecha en la cual se instala en la UDT.
- Observación del dispositivo: describe algún altercado al momento de crear, modificar, o eliminar el dispositivo del sistema.

3.2.8.2.Mantenimiento de dispositivos:

- Crear dispositivos
- Editar dispositivos
- Eliminar dispositivos
- Buscar dispositivos

3.2.9. GESTION DE USUARIOS

Las personas que usarán el software que se prevé sería en este caso la municipalidad y la entidad encargada del transporte público en la ciudad de Cuenca. Por lo cual necesitamos registrar esta información.

3.2.9.1.Información básica de usuarios:

- ID usuario: identificador único en el sistema
- Nombre del usuario: indica el nombre (login del usuario)
- Clave del usuario: indica la clave de acceso al sistema.
- Cedula del usuario: guarda el número de cedula del usuario.
- Nombres del usuario: almacena los nombres completos de la persona que será registrada como usuario.
- Tipo de usuario: indica si el usuario es administrador, o un usuario limitado.
- Observación del usuario: detalles que puedan ocurrir son registrados en este campo.

3.2.9.2.Mantenimiento de usuario:

- Crear usuario
- Editar usuario
- Eliminar usuario
- Inhabilitar usuario
- Buscar usuario

3.2.10. GESTION DE RUTAS

3.2.10.1. *Información básica de las rutas:*

PROPIA

- ID ruta: identificador único en el sistema
- Nombre de la ruta: nombre descriptivo acerca de la ruta que pueden realizar las UDTs
- Línea de la ruta: indica la línea de la ruta.
- Nombre operadora de la ruta:
- Condiciones de la flota de la ruta
- Id cooperativa: identificar de la cooperativa a la cual pertenece la ruta.

EXTERNA

- ID recorridos: identificador único en el sistema
- Nombre de la calle de ida: cada ruta está conformada por varios recorridos. Aquí se describe el nombre de la calle, de ida.
- Nombre de la calle de vuelta: cada recorrido se da tanto de ida como de vuelta, es por eso que aquí se registra el nombre de la calle cuando la UDT está de vuelta.
- ID ruta: indica la ruta a la cual pertenece la ruta.

3.2.10.2. *Mantenimiento de ruta*

- Crear ruta
- Editar ruta
- Eliminar ruta
- Inhabilitar ruta
- Buscar ruta
- Mantenimiento de los recorridos
 - Crear recorrido
 - Editar recorrido
 - Eliminar recorrido
 - Inhabilitar recorrido
 - Buscar recorrido

3.2.11. GESTION DE REPRESENTANTES

Cada cooperativa se supone tiene un representante legal, al cual contactar por cualquier eventualidad por lo cual esta información es relevante.

3.2.11.1. *Información básica de los representantes:*

- ID representante
- Cedula del representante
- Nombre del representante
- Dirección del representante
- Telefono1 del representante
- Telefono2 del representante
- Email del chofer
- ID de la cooperativa

3.2.11.2. *Mantenimiento de representantes:*

- Crear representante
- Editar representante
- Eliminar representante
- Inhabilitar representante
- Buscar representante

Una vez que hemos definido las funciones que tendrá el Sistema de control sonoro comenzaremos la parte del diseño para lo cual se desarrollan a continuación cada uno de los diagramas requeridos.

Comenzamos con los diagramas estáticos.

DIARAMA DE CLASES DEL SISTEMA DE CONTROL SONORO

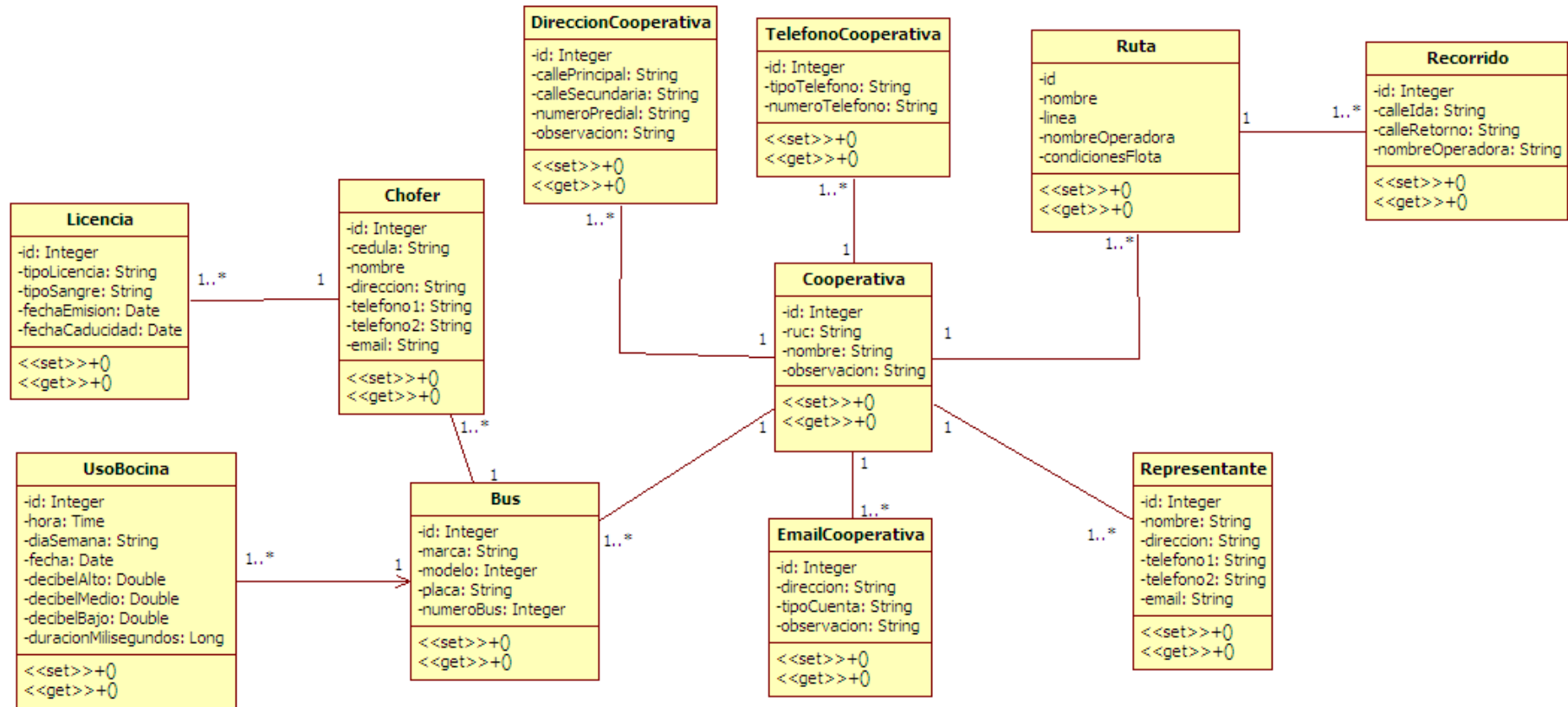


Figura 22 Diagrama de clases del SCS.

3.3. CASOS DE USO DEL SISTEMA DE CONTROL SONORO

3.3.1. USUARIOS

Actores identificados:

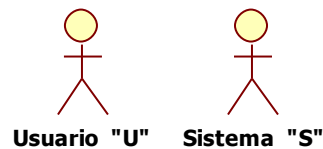


Figura 23 Actores Identificados CU. Usuarios

3.3.1.1. *Dar de alta un usuario:*

Permite registrar los datos de un usuario nuevo al sistema

Prerrequisitos

Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.

Flujo Básico

- S: muestra la pantalla necesaria para el caso (nombre del usuario, login, clave, etc.).
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos de los usuarios. En caso de que los datos sean incorrectos E1, E2, E3.

A2 S: sale de la ventana sin guardar los cambios

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.1.2. Editar un usuario:

Permite editar ya sea agregar, eliminar o modificar los datos de un usuario ya existente.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Estar registrado el usuario al cual se quiere editar.

Flujo Básico

- S: muestra la pantalla necesaria para el caso y permite la búsqueda y selección del usuario.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos de los usuarios. En caso de que los datos sean incorrectos E1, E2, E3.

A2 S: sale de la ventana sin guardar los cambios

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.1.3. Borrar un usuario: permite eliminar del sistema a los usuarios registrados.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Existir el usuario registrado en el sistema.

Flujo Básico

- S: muestra la pantalla necesaria para buscar y seleccionar el usuario.

- U: escoge al usuario que desea eliminar.
- S: muestra las opciones A1(Eliminar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que se haya seleccionado un usuario
- S: Si el usuario existe el sistema elimina a mencionado usuario siempre que el usuario que elimina no sea el mismo que está conectado En caso de que los no se elimina E1, E2.

A2 S: sale de la ventana sin guardar los cambios

Flujos de Excepción

- E1 S: el sistema no puede eliminar el usuario verifique que tenga los permisos necesarios para esta acción.
- E2 S: No se puede eliminar a sí mismo.

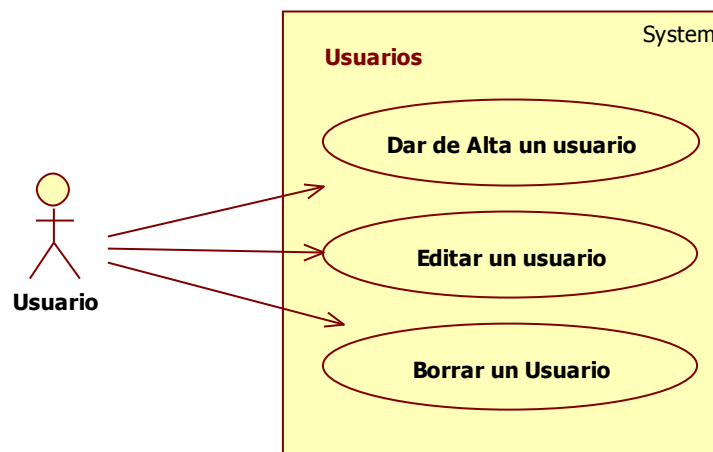


Figura 24 Caso de Uso Usuarios

3.3.2. COOPERRATIVAS

Actores identificados:

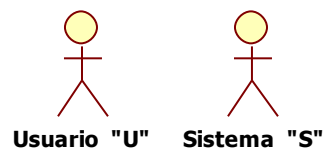


Figura25 Actores identificados en el caso de Uso Cooperativas

3.3.2.1. Dar de alta una cooperativa

Registra los datos de las cooperativas de transporte público.

Prerrequisitos

Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id,ruc, nombre, etc.)
- U: ingresa los datos requeridos por el sistema(Ver caso de uso Direcciones de cooperativas, Teléfonos de cooperativas , Email de cooperativas)
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos correctos el sistema almacena en la base de datos los datos de las cooperativas. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.2.2. Editar una cooperativa.

Permite agregar, borrar o editar los datos de una cooperativa que ya haya sido ingresada anteriormente.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber ingresado la cooperativa a editar

Flujo Básico

- S: muestra la pantalla necesaria para seleccionar la cooperativa que se desee editar.

- U: una vez seleccionado la cooperativa, ingresa los datos requeridos por el sistema(Ver caso de uso Direcciones de cooperativas, Teléfonos de cooperativas , Email de cooperativas)
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos correctos el sistema almacena en la base de datos los datos de las cooperativas. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

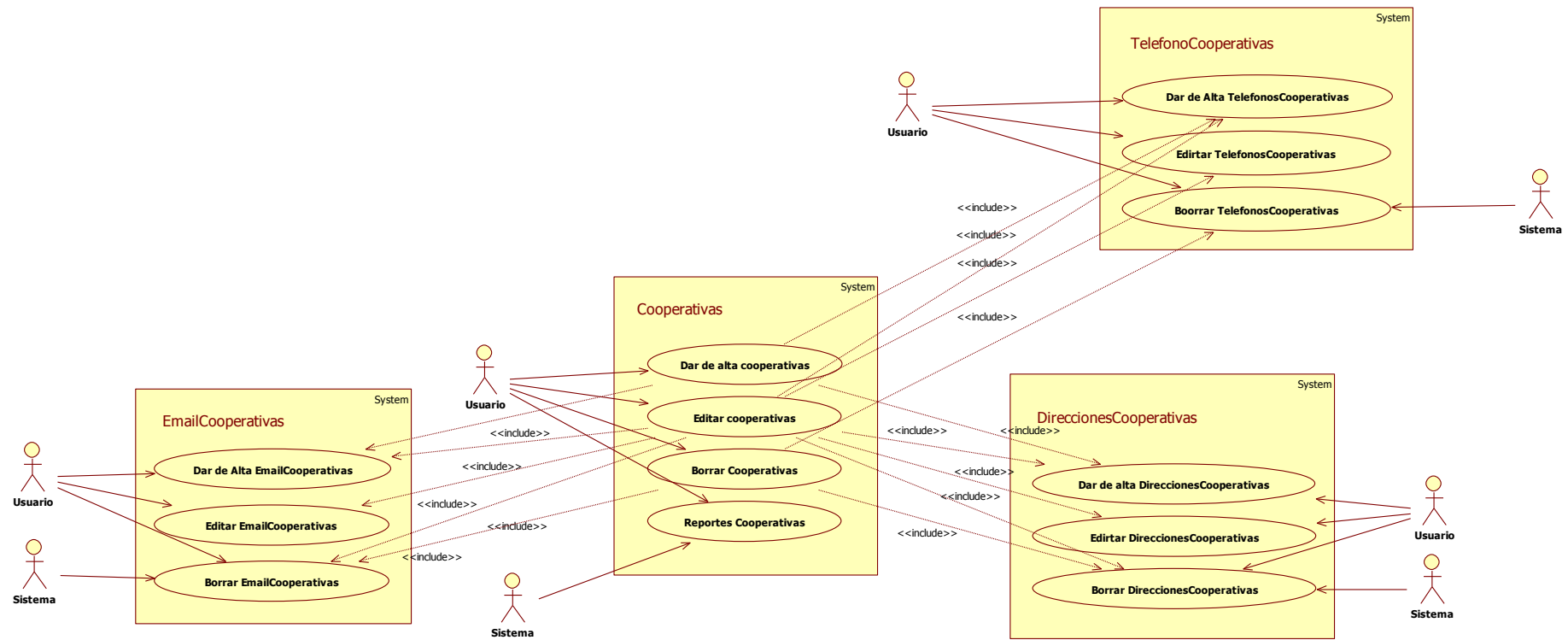


Figura 26 Diagrama de caso de uso de las cooperativas

3.3.3. BUSES

Actores identificados:

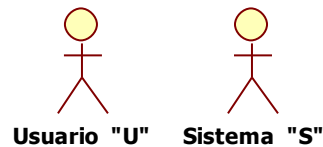


Figura 27 Actores Caso de Uso de Buses

3.3.3.1. *Dar de alta un bus.*

Registra los datos de las unidades de transporte público.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id, marca, modelo, etc.) ver caso de uso buscar y seleccionar cooperativa.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos del bus ingresado. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.3.2. *Editar un bus*

Permite agregar, eliminar, y modificar los datos de las unidades de transportes.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber registrado anteriormente el bus en el sistema

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id, marca, modelo, etc.)ver caso de uso buscar y seleccionar cooperativa.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos de los buses. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.3.3.Eliminar un bus

Permite eliminarlos datos de las unidades de transporte público.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber registrado el bus en el sistema

Flujo Básico

- S: muestra la pantalla necesaria para el caso permitiendo al usuario seleccionar la UDT que desee eliminar.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(eliminar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que el usuario tenga los permisos para realizar la acción. En caso de que los datos sean incorrectos E1, E2, E3.

A2 S: cierra la ventana si guardar los cambios

Flujos de Excepción

E1 S: el sistema indica que no tiene los permisos necesarios.

E2 S: muestra un mensaje indicando que no se puede eliminar a sí mismo.

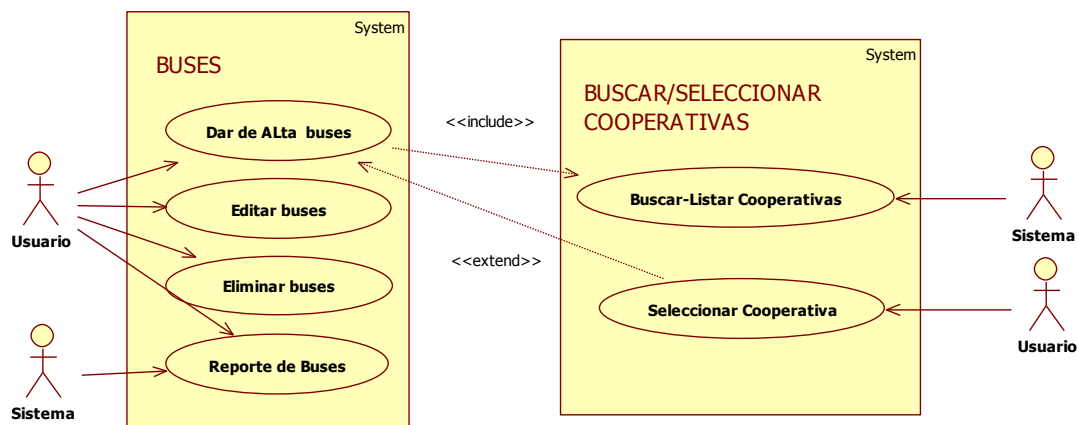


Figura 28 Diagrama de caso de uso de Buses

3.3.4. RUTAS

Actores identificados:

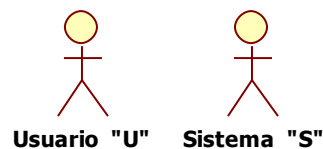


Figura 29 Actores Caso de Uso de Rutas

3.3.4.1. Dar de alta una ruta.

Registra los datos de las rutas que recorren las unidades de transporte público.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id,nombre ruta, etc.) ver caso de uso buscar y seleccionar cooperativa.Y caso de uso Recorridos.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos de la ruta ingresada. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.4.2.Editar una ruta:

Permite agregar, eliminar, y modificar los datos de las rutas que recorren las unidades de transportes.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber registrado anteriormente la ruta en el sistema

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id,nombre ruta, etc.)ver caso de uso buscar y seleccionar cooperativa. Y caso de uso Recorridos
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).

- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos de los buses. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.4.3. Eliminar una ruta

Permite eliminar los datos de las rutas que recorren las unidades de transporte público.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber registrado la ruta en el sistema

Flujo Básico

- S: muestra la pantalla necesaria para el caso permitiendo al usuario seleccionar la ruta que desee eliminar.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(eliminar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que el usuario tenga los permisos para realizar la acción. En caso de que los datos sean incorrectos E1, E2, E3.

A2 S: cierra la ventana si guardar los cambios

Flujos de Excepción

- E1 S: el sistema indica que no tiene los permisos necesarios.

E2 S: muestra un mensaje indicando que no se puede eliminar a sí mismo.

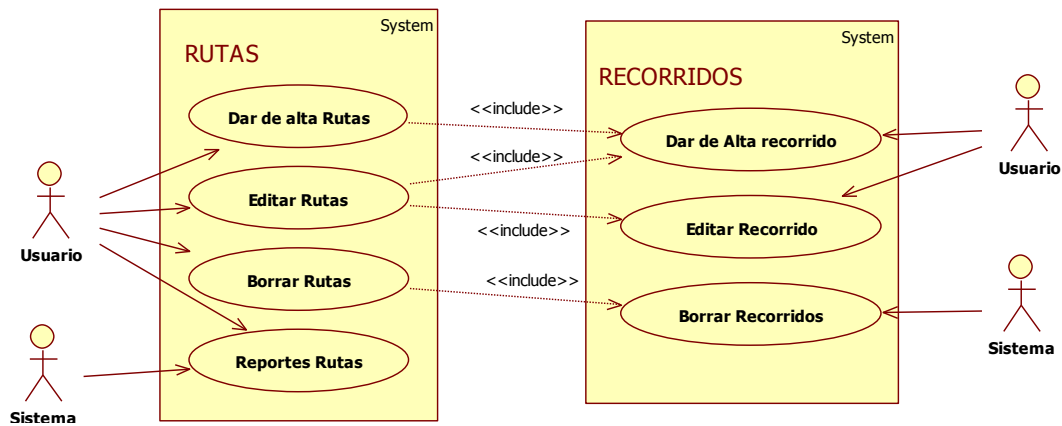


Figura 30 Diagrama de caso de uso de Rutas

3.3.5. CHOFERES

Actores identificados:

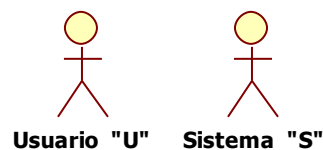


Figura 31 Actores Caso de Uso de Choferes

3.3.5.1. Dar de alta un chofer.

Registra los datos de los choferes que conducen las unidades de transporte público.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id, nombre cedula, etc.) ver caso de uso buscar y seleccionar Buses. Y caso de uso Licencias.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos del chofer. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.
- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.5.2. Editar un chofer

Permite agregar, eliminar, y modificar los datos de los choferes que conducen las unidades de transportes.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber registrado anteriormente la ruta en el sistema

Flujo Básico

- S: muestra la pantalla necesaria para el caso (id, nombre cedula, etc.)ver caso de uso buscar y seleccionar Bus. Y caso de uso Licencias
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(guardar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1

- S: El sistema verifica que todos los datos requeridos sean ingresados.
- S: Si los datos son correctos el sistema almacena en la base de datos los datos del chofer. En caso de que los datos sean incorrectos E1, E2, E3.

Flujos de Excepción

- E1 S: el sistema indica que faltan datos.

- E2 S: muestra un mensaje indicando que los datos están incorrectos.
- E3 S: el sistema indica que hubo un error al momento de almacenar los datos.

3.3.5.3. Eliminar un chofer

Permite eliminar los datos del chofer conduce las UDT.

Prerrequisitos

- Estar registrado en el sistema y tener los permisos necesarios para la ejecución del caso de uso.
- Haber registrado al chofer en el sistema

Flujo Básico

- S: muestra la pantalla necesaria para el caso permitiendo al usuario seleccionar el chofer que desee eliminar.
- U: ingresa los datos requeridos por el sistema.
- S: muestra las opciones A1(eliminar) y A2(salir).
- U: escoge una de las opciones A1 o A2.

Flujo Alternativo

A1 S: El sistema verifica que el usuario tenga los permisos para realizar la acción. En caso de que los datos sean incorrectos E1, E2, E3.

A2 S: cierra la ventana si guardar los cambios

Flujos de Excepción

- E1 S: el sistema indica que no tiene los permisos necesarios.
- E2 S: Indica que no se puede eliminar a sí mismo.

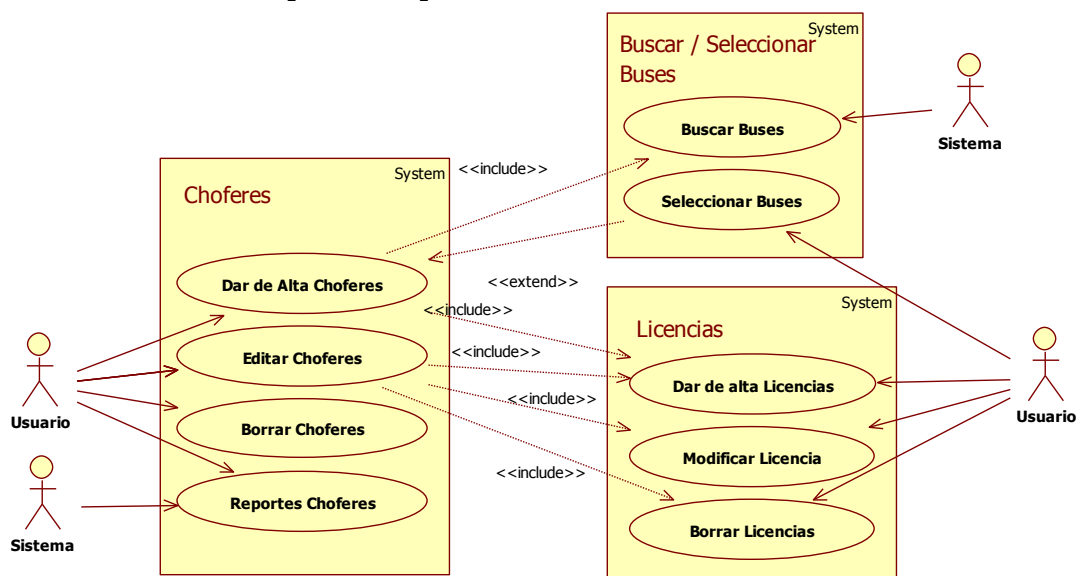


Figura 32 Caso de uso de choferes

CAPITULO 4

4. PROGRAMACION E IMPLEMENTACION

4.1. Introducción

Hemos llegado a nuestro último capítulo, todo proyecto contiene varias etapas en el proceso de desarrollo, en capítulos anteriores hablamos sobre el análisis y diseño de la aplicación, a más de la estructura del sistema de control sonoro, en el presente capítulo se detalla el proceso de desarrollo del software así como la fase de implementación.

En el proceso de desarrollo se estipula todo lo concerniente al desarrollo de software llamado también programación de la aplicación, por lo cual se requiere de conocimientos previos sobre el lenguaje de programación y el IDE (entornos de desarrollo a usar), Actualmente las grandes consultoras, tienen el conocimiento previo sea en el lenguaje de programación así como en el entorno de desarrollo, iniciar un proyecto en el que intervengan lenguajes o IDE's desconocidos para la empresa consultora significaría grandes pérdidas económicas ya que esto involucra un tiempo de aprendizaje de dichas herramientas.

Para el sistema de control sonoro se ha considerado la investigación como parte esencial del proyecto ya que los conocimientos previos tanto del lenguaje de programación como el IDE, no existían. Por lo cual se empleó gran parte del tiempo a la investigación y el aprendizaje.

La fase de implementación también tuvo sus complicaciones ya que es una etapa crítica en la cual involucra al usuario final quien es el que generalmente encuentra los errores en las aplicaciones. Por eso esta etapa también incluye detección y corrección de errores.

4.2. FUNDAMENTOS DEL LENGUAJE DE PROGRAMACION

4.2.1. Microsoft Visual Studio

Es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Adicionalmente, Microsoft ha puesto gratuitamente a disposición de todo el mundo una versión reducida de MS SQL Server llamada SQL Server Express Edition cuyas principales limitaciones son que no soporta bases de datos superiores a 4 GB de tamaño, únicamente utiliza un procesador y 1 Gb de Ram, y no cuenta con el Agente de SQL Server.

Para el desarrollo del sistema de control sonoro se utilizó el lenguaje de programación Visual Basic .NET

LINQ (LanguageIntegratedQuery) es un nuevo conjunto de herramientas diseñado para reducir la complejidad del acceso a Base de Datos, a través de extensiones para C++ y Visual Basic así como para Microsoft .NET Framework. Permite filtrar, enumerar, y crear proyecciones de muchos tipos y colecciones de datos utilizando en todos la misma sintaxis, prescindiendo del uso de lenguajes especializados como SQL o XPath.

Para el manejo de datos hemos adoptado la nueva tecnología que ofrece Microsoft para el manejo de datos esto es ADO. Entity Framework, si quisiéramos comparar con las tecnologías open source tiene gran similitud con el manejo de JPA para el desarrollo en aplicaciones JAVA. Esta extensión la podemos obtener desde el NET Framework 3.5 (incluido en el service pack 1 de Visual Studio 2008).

4.2.2. ADO Entity Framework

La nueva herramienta para el manejo de datos nos facilita el todo el trabajo. A continuación se indica de manera muy general el manejo de esta herramienta.

Ingresamos a Visual Studio y creamos una aplicación de formularios con el lenguaje Visual Basic

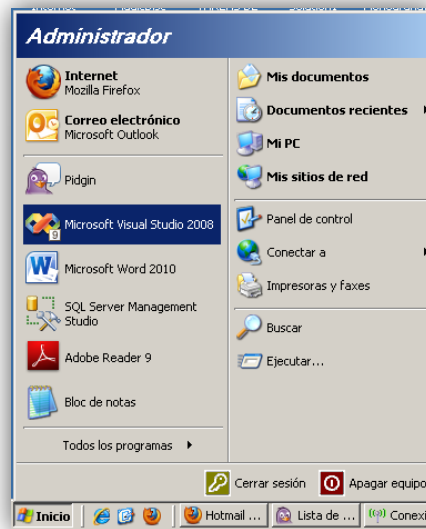


Figura 33 Ingreso a Visual Studio

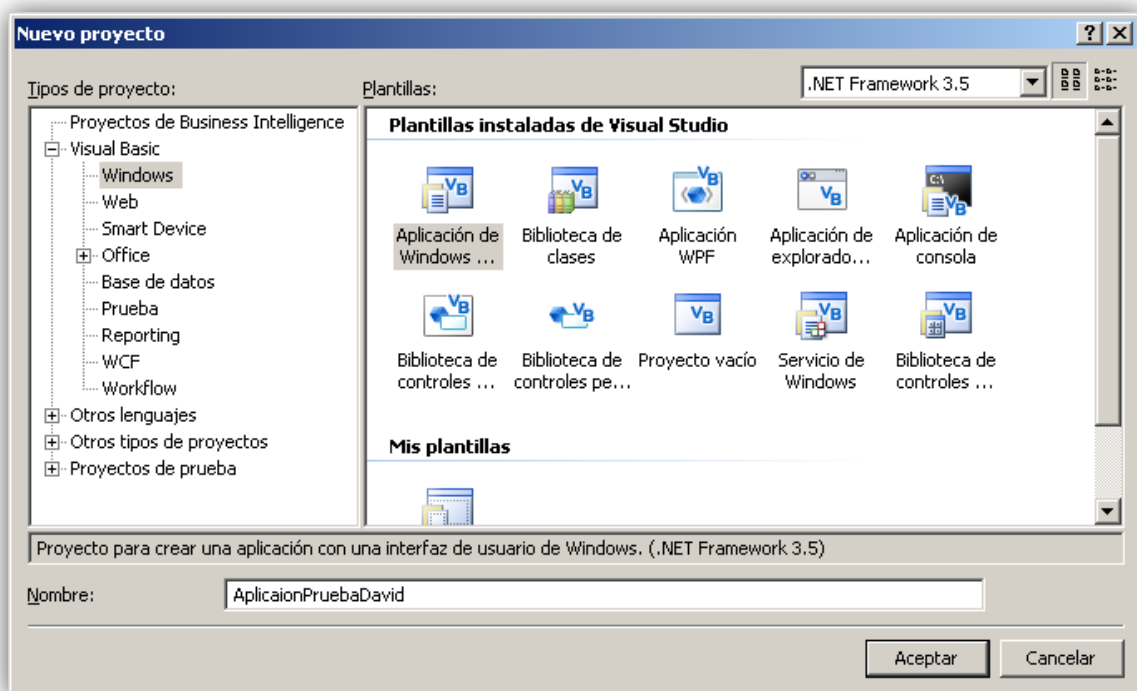


Figura 34 Creación de una aplicación de prueba

Creamos el modelo Entity que nos permitirá manejar los datos. Para lo cual vamos al explorador de soluciones y damos clic con el botón secundario del ratón y ponemos agregar nuevo elemento.

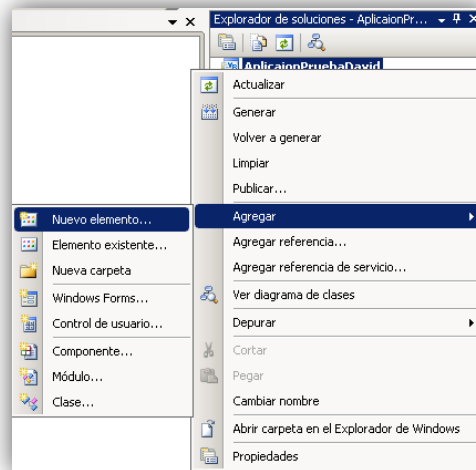


Figura 35 Agregar un nuevo elemento

Escogemos el ADO.NET Entity Data Model dentro de las plantillas instaladas de Visual Studio. Y damos clic en Agregar.

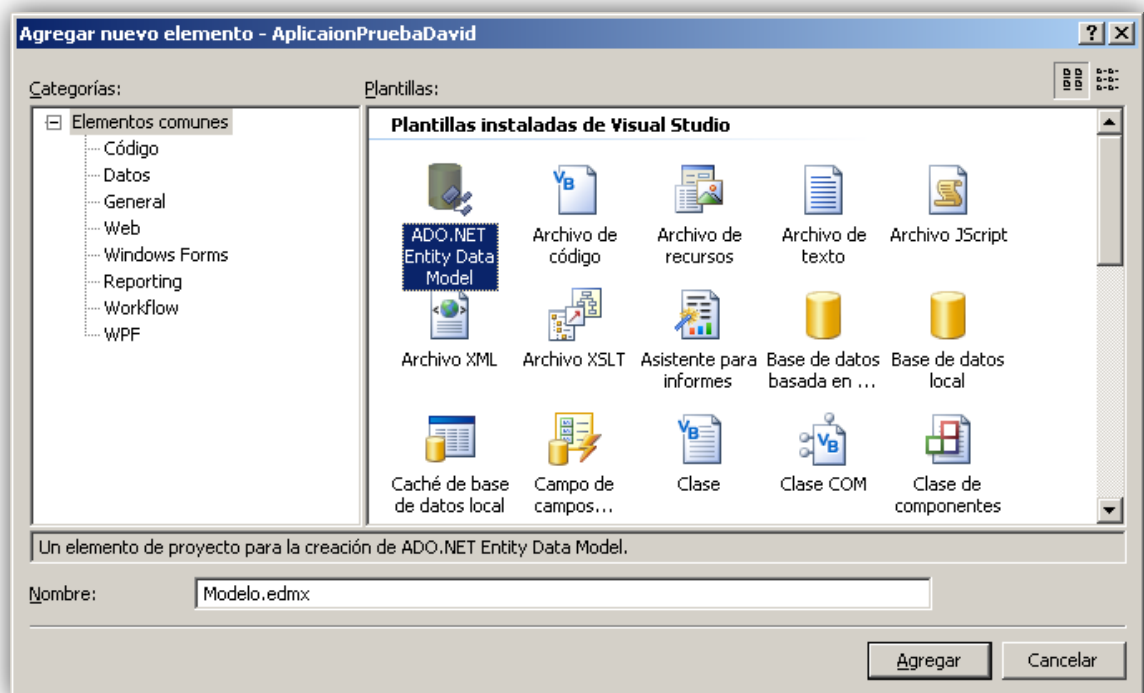


Figura36 ADO.NET Entity Data Model

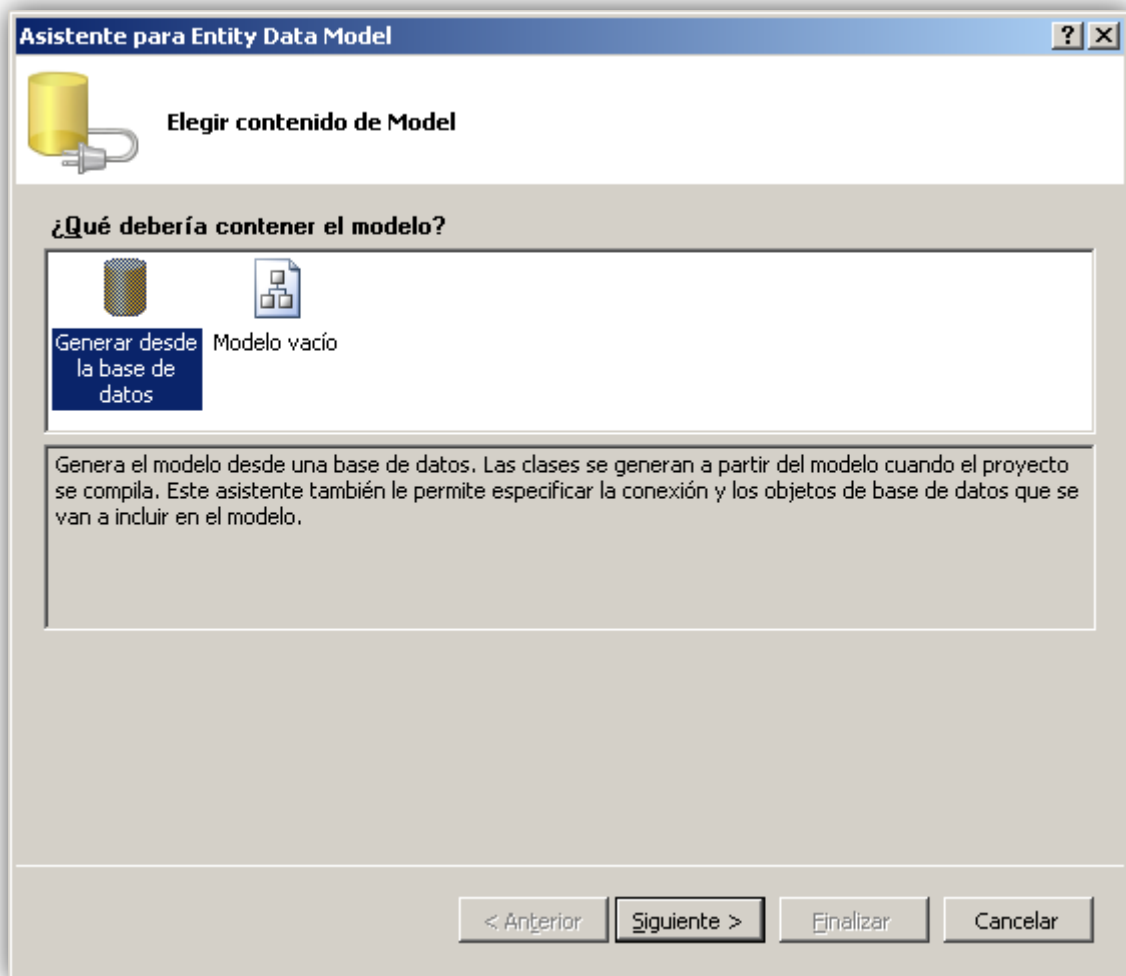


Figura 37 Creación del modelo a partir de una base de datos existente

Podemos crear un modelo vacío o desde una base de datos, para la prueba usaremos una tabla de una base de datos ya creada.

Seleccionamos o creamos una conexión nueva con la base de datos. En este caso seleccionaremos una conexión existente. Le asignamos un nombre y damos clic en siguiente.

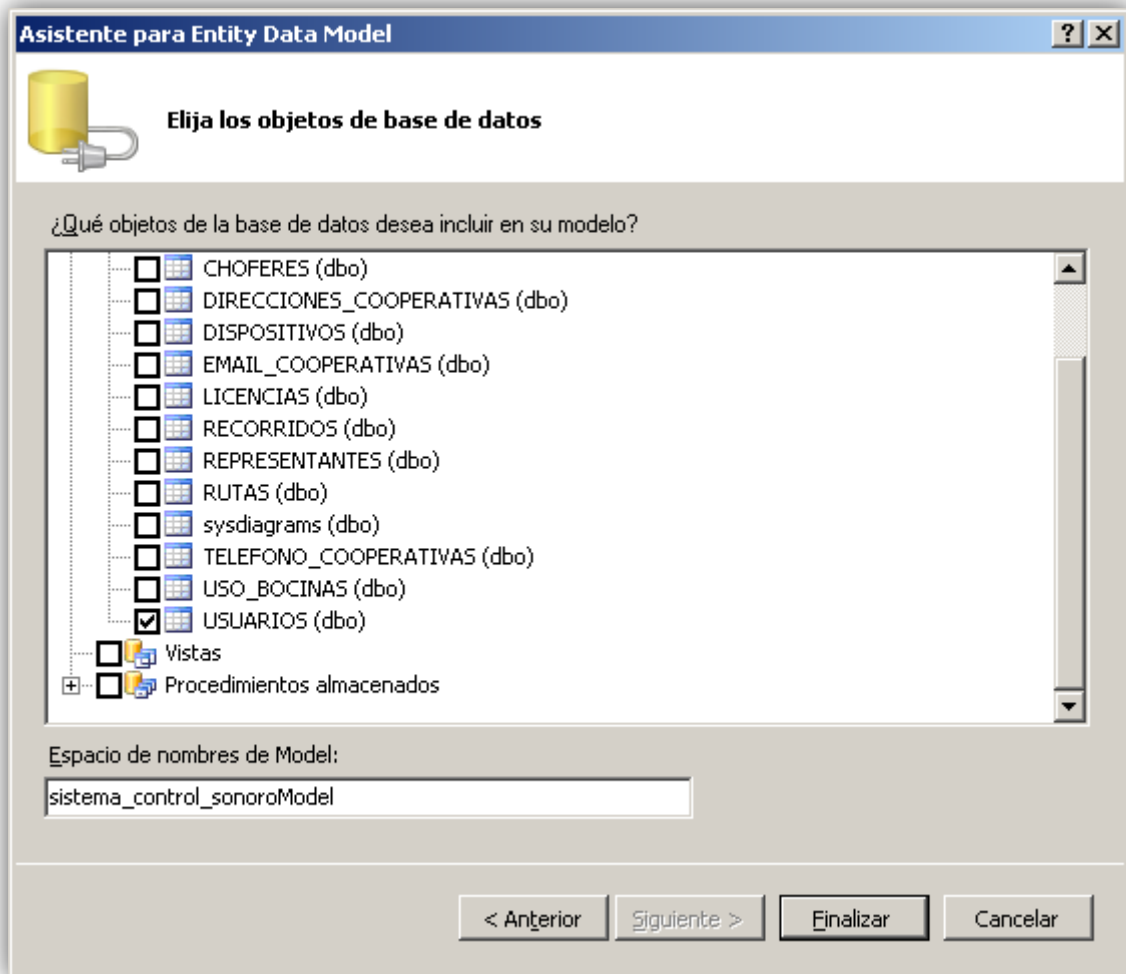


Figura 38 Selección de la tabla usuarios

Escogemos las tablas que sean necesarias en este caso solo escogeremos la tabla de usuarios. Damos clic en Finalizar y nuestro modelo se ha creado

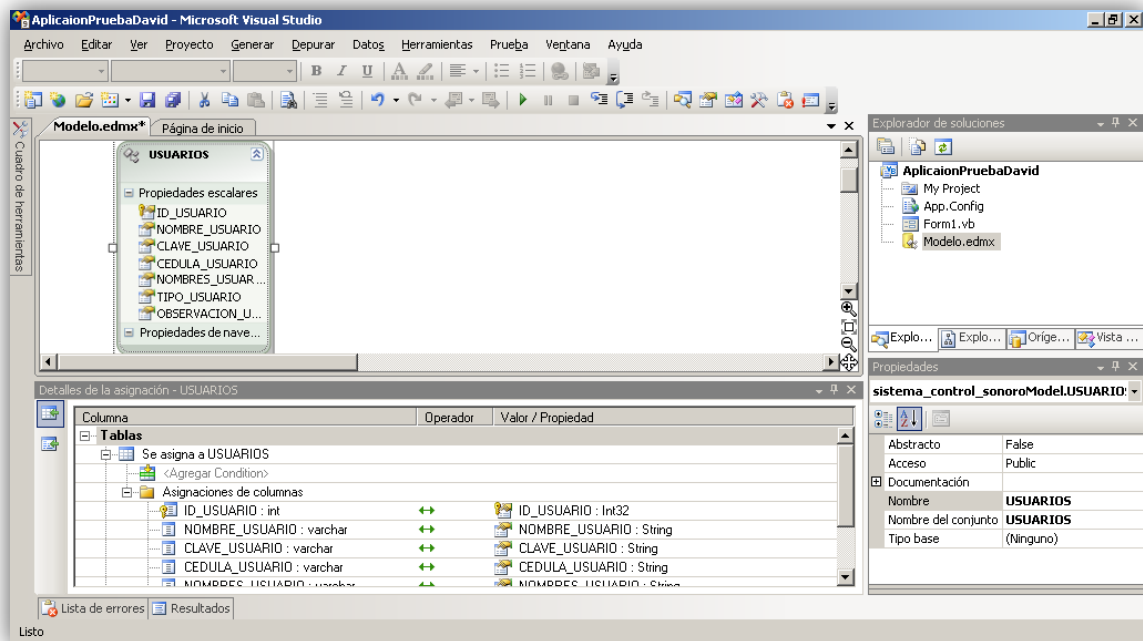


Figura 39 Modelo de la tabla Usuarios

Una vez creado el modelo el manejo de los datos es sencillo, para esto creamos un formulario con dos campos nada mas

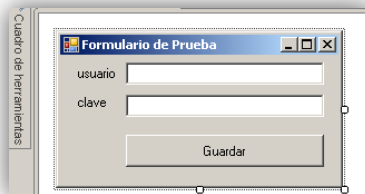


Figura 40 Creación del formulario de prueba dos campos

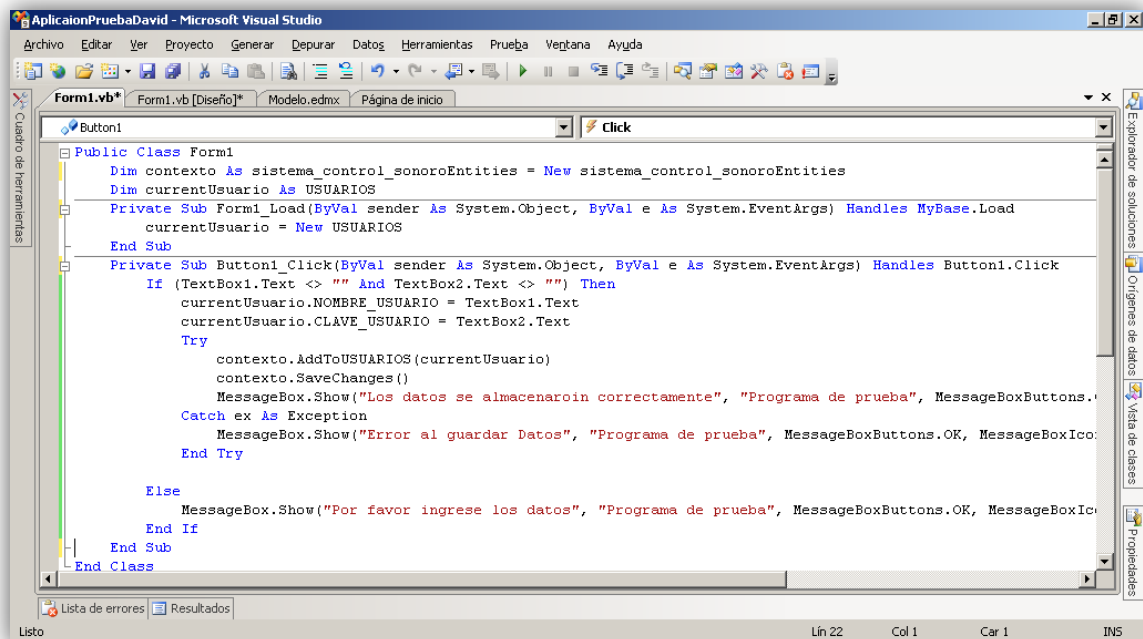


Figura 41 Código para manejo de datos con EF y Visual Basic .NET

Y vamos al código. Como se observa el almacenamiento de los datos es sencillos solo nos referimos a dos métodos del contexto una el agregar el currentUsuario y la otra saveChanges.

Al ejecutar la aplicación tendremos lo siguiente.

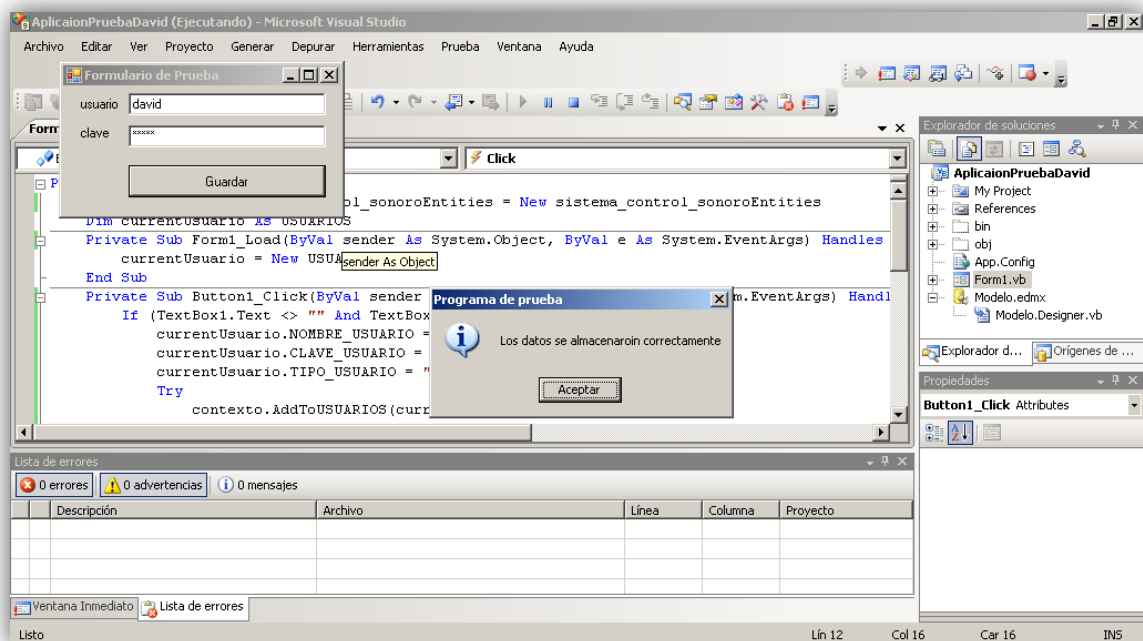


Figura 42 Aplicación de prueba en ejecución

4.3. Desarrollo del Sistema de control sonoro

Por lo cual se requiere de un diagrama de entidades que la misma herramienta nos permite realizar

Aunque ya existen versiones más recientes para el desarrollo del sistema se empleó la versión 3.5 propia del Net Framework que se utilizó.

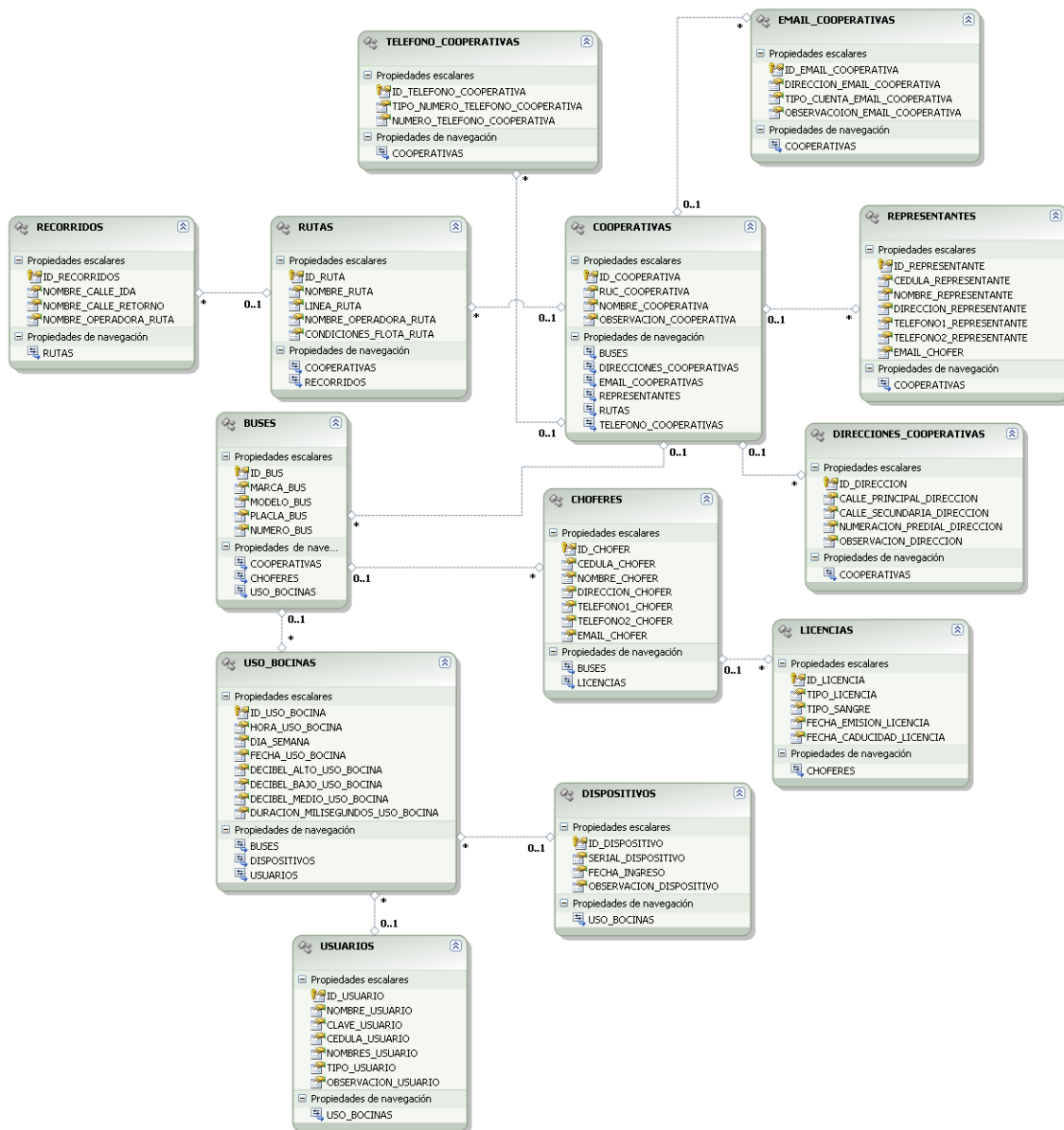


Figura 43 Diagrama de Entidades dentro de Visual Studio

4.4. Base de datos:

Cualquier lenguaje que se haya escogido se integra perfectamente con MS SQL Server, como motor de la Base de datos. Es por eso que se utilizó este motor de base de datos.

En capítulos anteriores se trataron los diagramas de U.M.L. pero ahí no se incluyó el diagrama ER.¹⁶, el cual nos indica cómo van a estar relacionados los datos en la base de datos.

4.4.1. Diagrama Entidad Relación:

“El Modelo Entidad Relación (ER) permite desarrollar un diseño de base de datos en un esquema de alto nivel conceptual sin considerar los problemas de bajo nivel como la eficiencia, el modelo implícito del administrador de base de datos o las estructuras físicas de los datos.”¹⁷.

En la siguiente figura se muestra el modelo para la BD que vamos a usar:

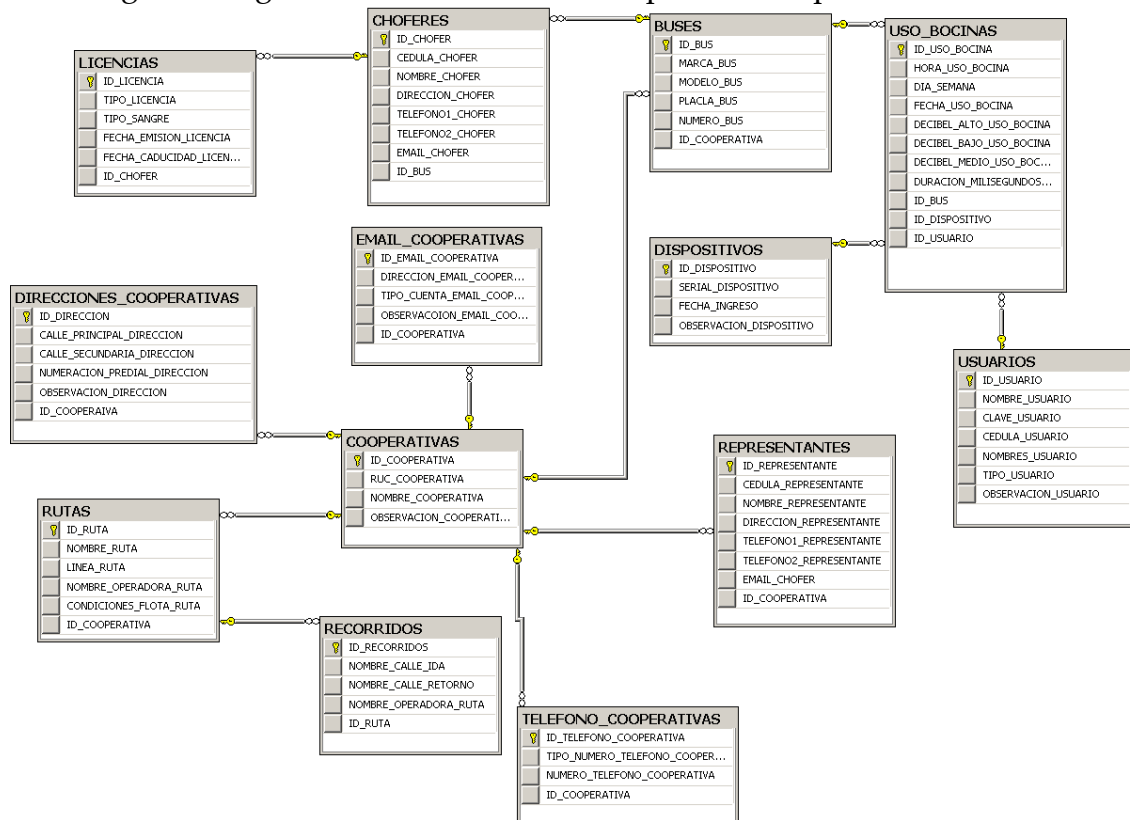


Figura 4445 Modelo de la base de datos para el SCS

¹⁶ER Entidad Relaciones una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información así como sus interrelaciones y propiedades.

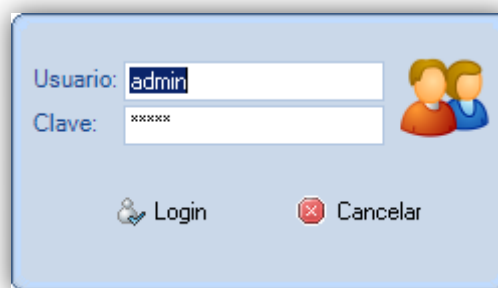
¹⁷Departamento de Ciencias de la Computación, Universidad de Nebraska de Omaha. NE 68182-0500, USA

El sistema contiene un solo Módulo con varias funcionalidades, como se detalló en el capítulo anterior existen varios subsistemas que gestionan cada uno de los requerimientos del sistema.

Para lo cual se muestra a continuación el detalle de cada una de las interfaces que intervienen en cada una de las gestiones del software.

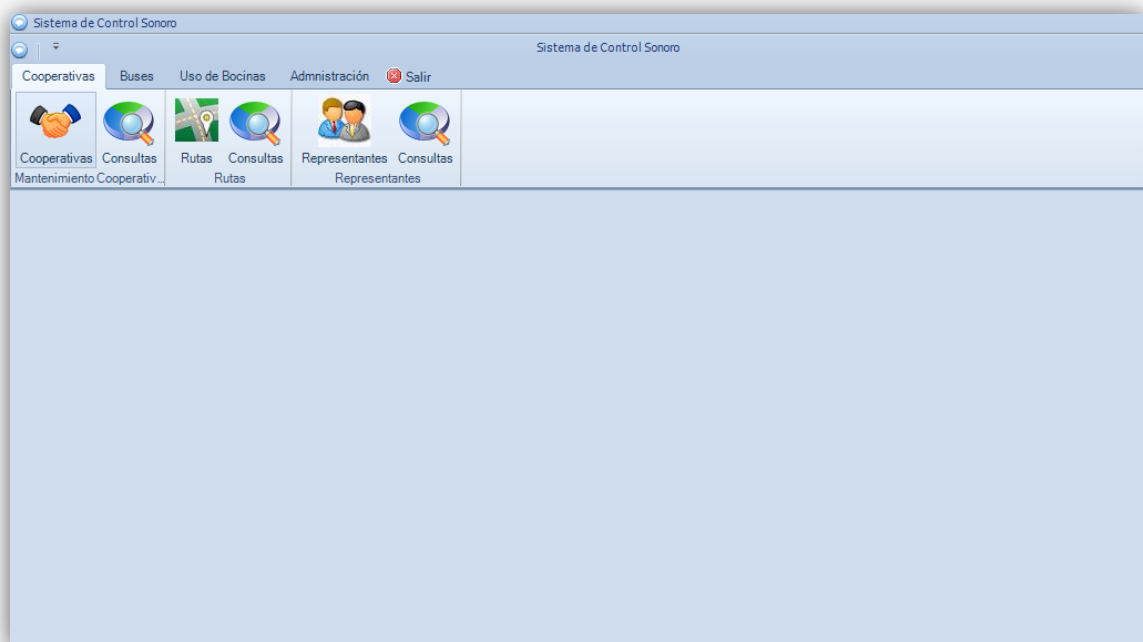
4.5. Mantenimiento de usuario:

En primera instancia nos muestra la ventana de ingreso en el cual debemos colocar el nombre de usuario (login) y la clave correspondiente.



Interfaz 1Login

El sistema verificará que los datos sean correctos, a continuación nos carga el programa

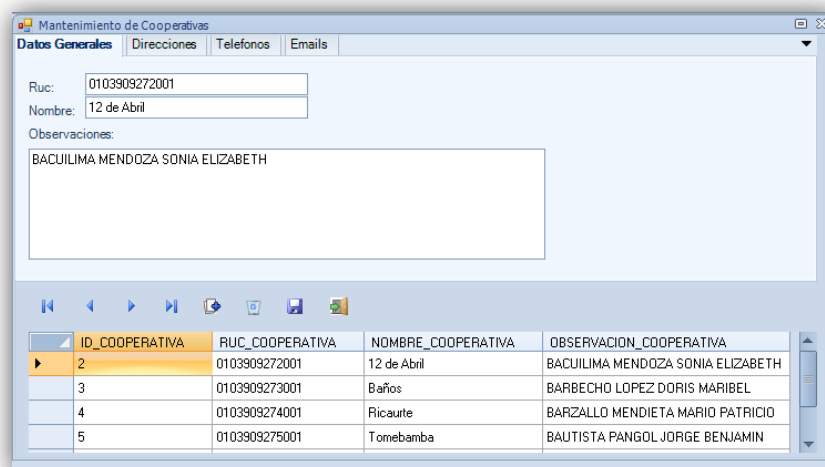


Interfaz 2 Ventana Principal

Como se observa en la imagen podemos gestionar todas las funciones anteriormente detalladas, comenzaremos por las cooperativas de transporte.

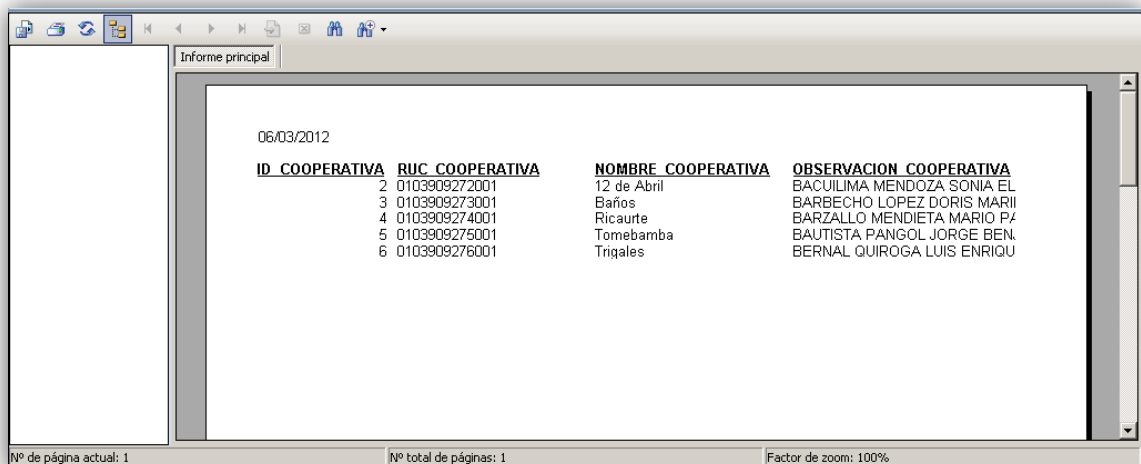
Al seleccionar la opción de cooperativas nos despliega la siguiente pantalla en la cual nos muestra los campos necesarios para registrar las cooperativas de transporte, a más de que contiene varias pestañas con información adicional como la posibles direcciones, números telefónicos de contactos y las posibles direcciones de correo electrónico.

El sistema contiene botones que facilita la navegación entre registros.



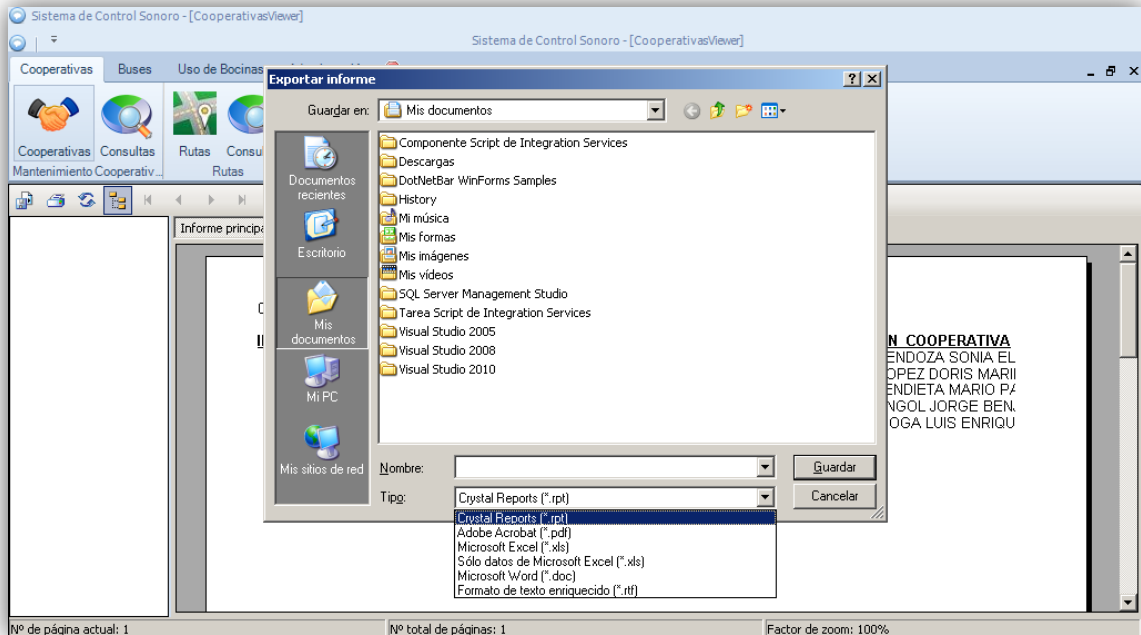
Interfaz 3 Cooperativas

El sistema nos permite generar un reporte con el listado de las cooperativas que se muestran en la grilla.



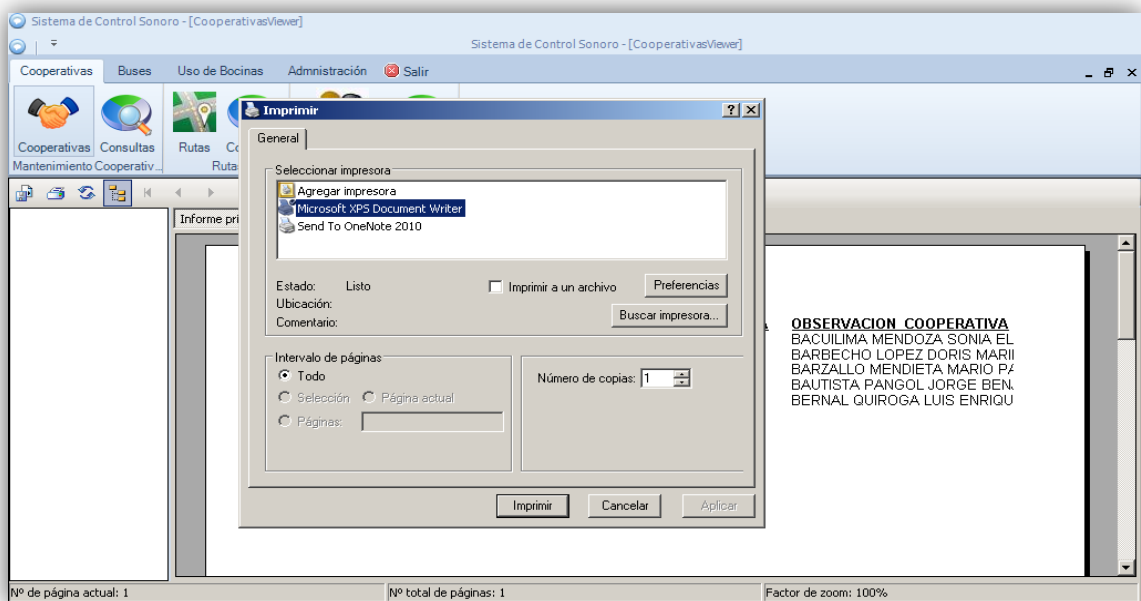
Interfaz 4 Reporte Cooperativas

La vista preliminar del reporte contiene varias opciones que nos pueden ayudar al momento de imprimir, como son imprimir, exportar , o configurar la página, a continuación se muestran cada una de las imágenes correspondientes.



Interfaz 5 Exportar Reporte Cooperativas

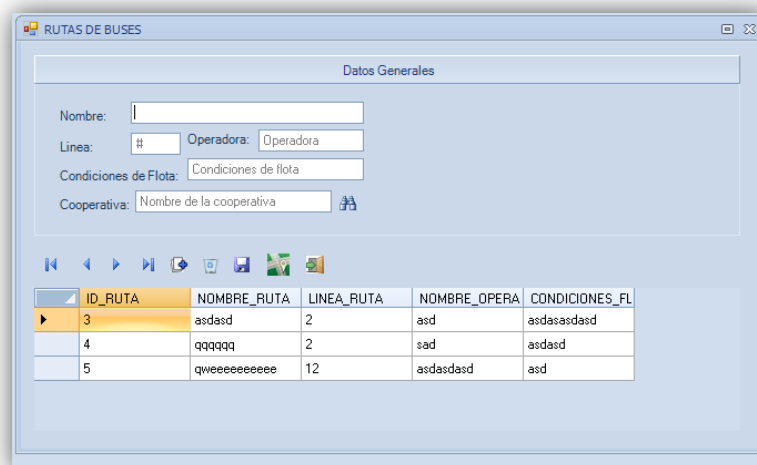
En la figura se muestran los formatos soportados para la exportación



Interfaz 6 Imprimir reporte de Cooperativas

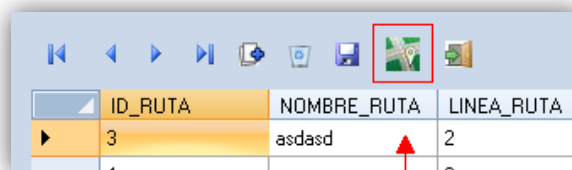
Nos despliega una ventana con las impresoras instaladas en el equipo con el cual podemos elegir la impresora que vamos a utilizar.

A continuación escogeremos la opción de las Rutas, como la mayoría de opciones son similares mostraremos exclusivamente la ventana principal de cada una de las funciones.



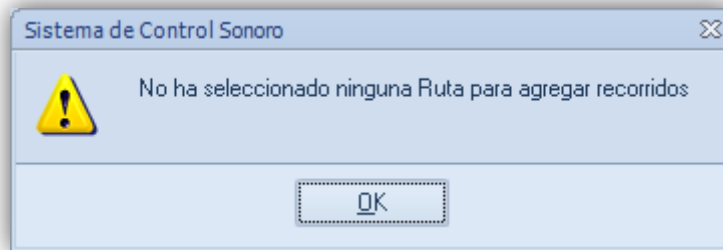
Interfaz 7 Buses

Como cada ruta tiene la opción de tener varios recorridos es decir varias calles que conforman la ruta, desde esta ventana podremos ingresar los recorridos, al dar clic en el botón



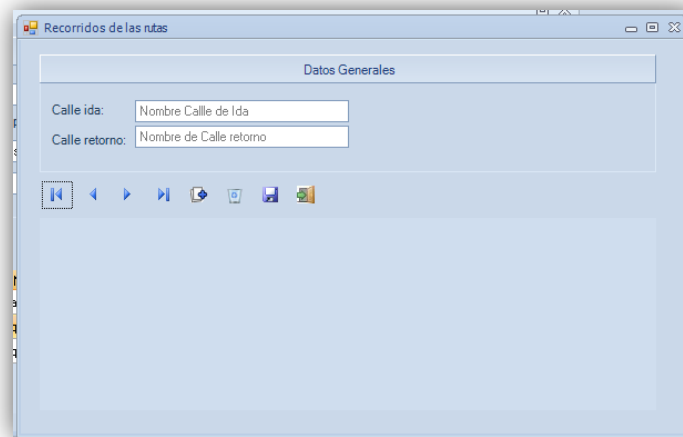
Interfaz 8 Botón recorrido

Nos despliega la siguiente ventana, en la cual podemos ir ingresando las calles que conforman la ruta, no existe un buscador de rutas para asociarlas desde cualquier recorrido lo que hace es desde la ruta que estemos ingresando le definimos por qué calles va a atravesar la ruta. Si no ha seleccionado una ruta previamente al tratar de ingresar en la pantalla de recorridos nos muestra el siguiente mensaje de advertencia.



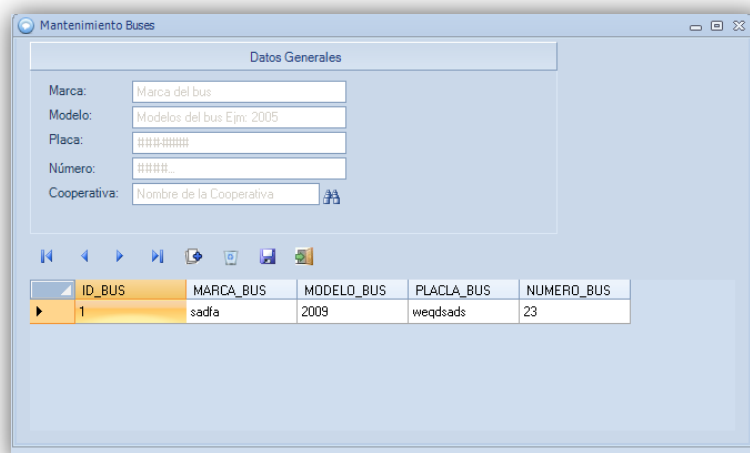
Interfaz 9 Mensaje de error al seleccionar la Ruta

En el caso de que ya haya seleccionado la ruta nos muestra una ventana como la siguiente imagen.

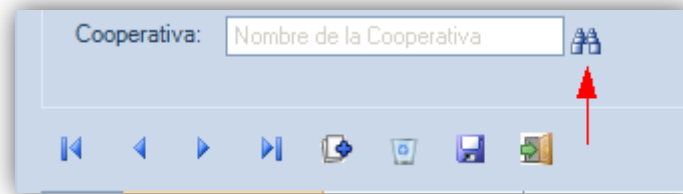


Interfaz 10 Recorridos

Continuamos con la ventana de Buses, uno de los campos involucra seleccionar una cooperativa por lo cual podemos buscar la cooperativa al dar clic en el botón

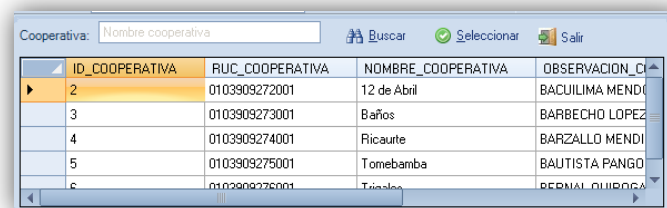


Interfaz 11 Buses



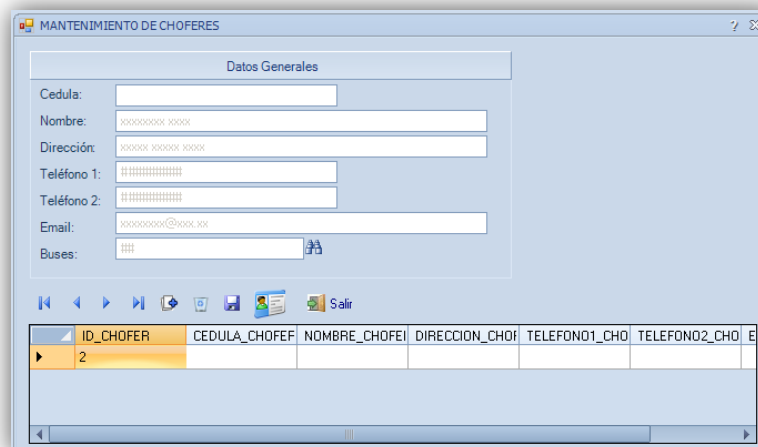
Interfaz 12 Botón Buscar Cooperativa

Nos muestra la siguiente ventana en la cual podemos buscar y seleccionar la cooperativa de transporte.



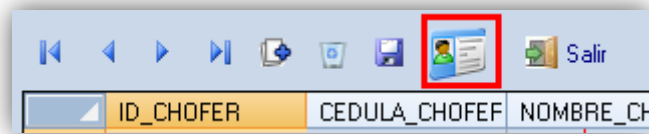
Interfaz 13 Buscar seleccionar cooperativas

Continuamos con el mantenimiento de Choferes

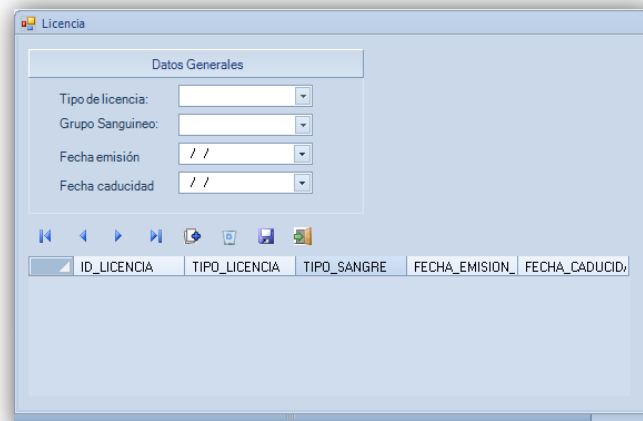


Interfaz 14 Choferes

Como sabemos un chofer puede tener varias licencias por lo cual podemos asignar directamente desde esta opción una o más licencias que tenga el chofer, al dar clic en el botón.



Interfaz 15 Botón licencia



Interfaz 16 Licencia

Básicamente ya nos hemos familiarizado con algunas de las interfaces del sistema de control sonoro.

Backup de la base de datos:

En un inicio no contamos con información relevante, es decir que se podría instalar el programa, y los datos se irían alimentando de a poco, sin embargo, a continuación se indica cómo sacar un backup de la base de datos MS SQL Server ya que éste es nuestro motor de base de datos.

Ingresamos al Microsoft SQL server Management Studio, y nos logeamos con nuestro nombre y usuario de la base de datos.

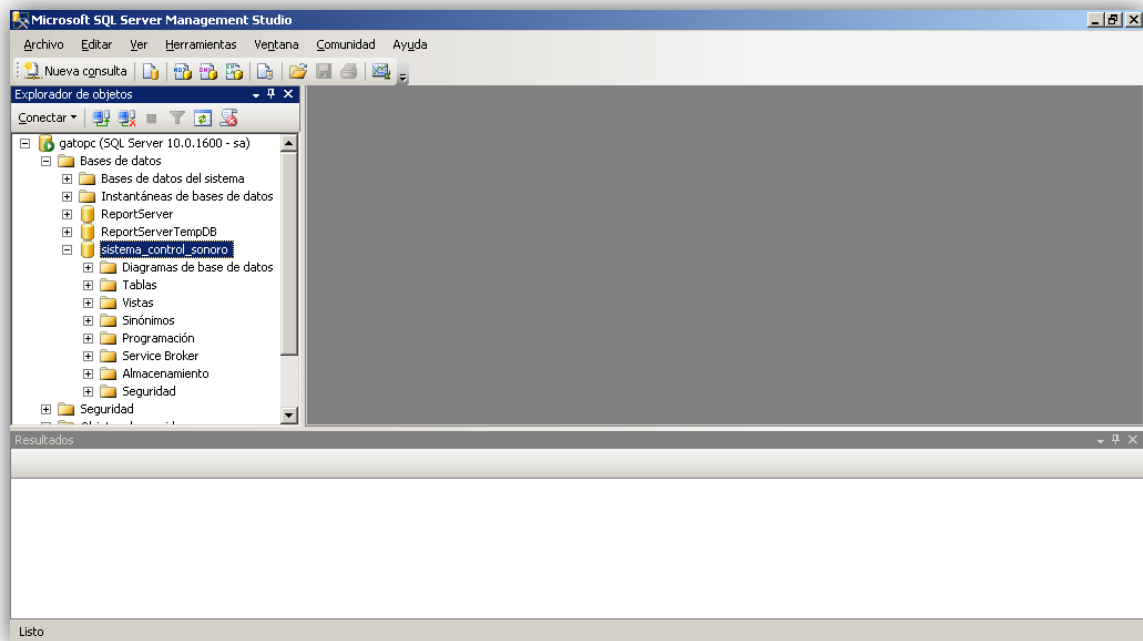


Figura46Microsoft SQL server Management Studio

Nos ubicamos sobre la base de datos que queremos sacar un backup, luego damos clic con el botón secundario sobre la base que elegimos y nos vamos a copia de seguridad.

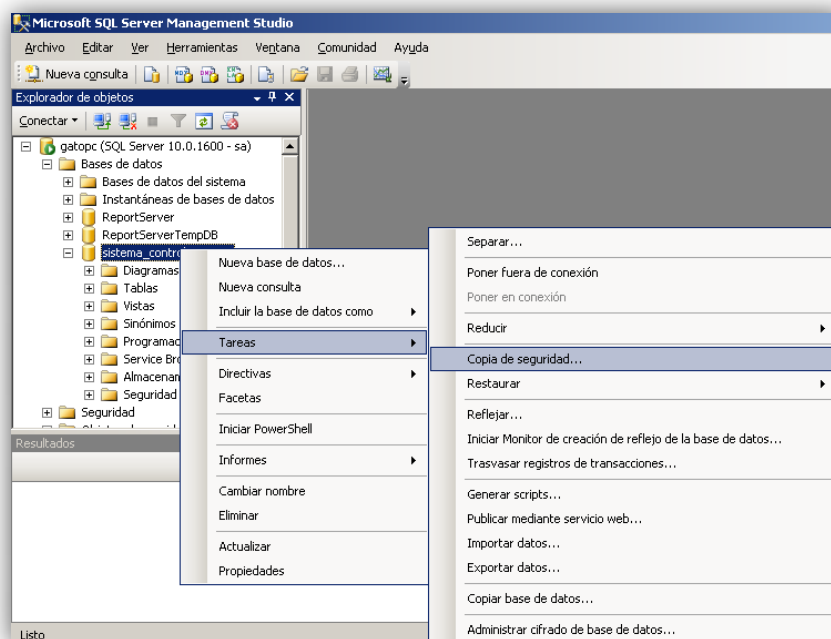


Figura 47 Crear copia de seguridad

Al seleccionar la opción anterior nos muestra una imagen como la siguiente

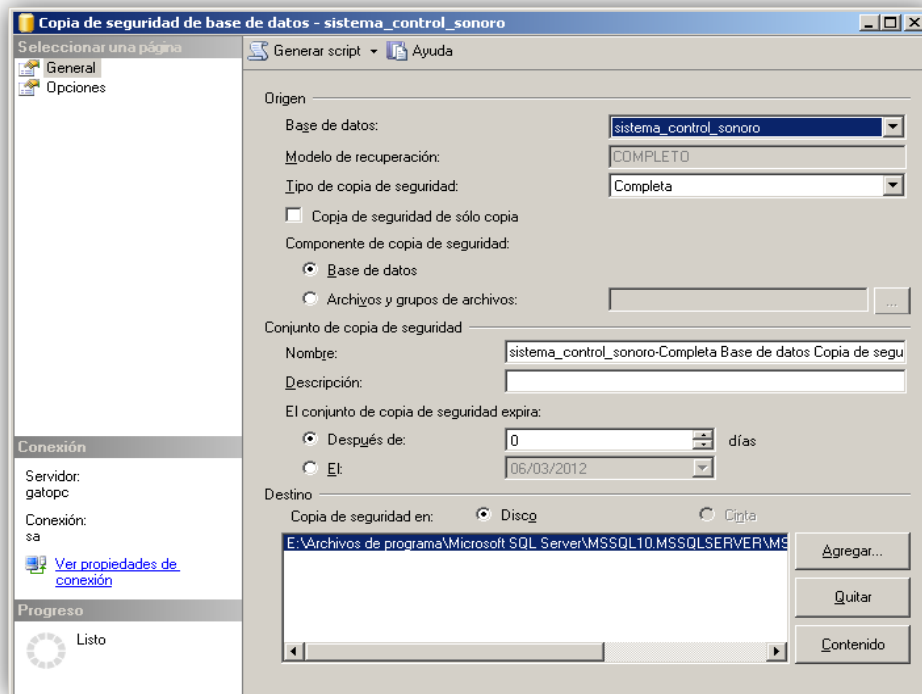


Figura 48 Ubicación de guardado de la base de datos

Verificamos que todos los datos sean correctos y damos clic en agregar luego en aceptar. Si todo salió bien nos muestra un mensaje

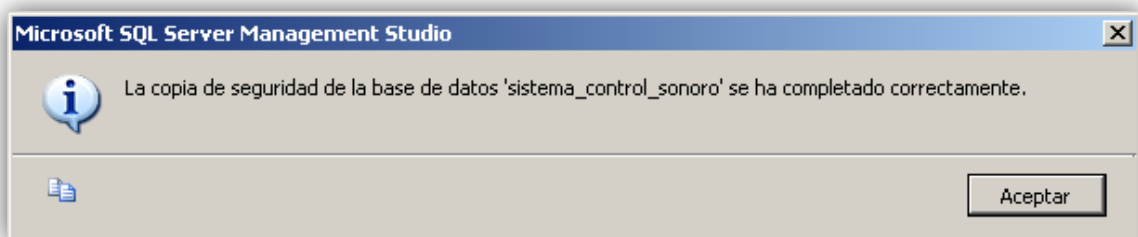


Figura 49 Mensaje de creación de respaldo de Base de datos

Una vez que tengamos el respaldo y necesitemos restaurar nuestra base, hacemos lo siguiente.

Ingresamos y nos logeamos (Ver pasos anteriores).

Damos clic con el botón secundario del ratón donde dice Base de Datos y nos muestra lo siguiente;

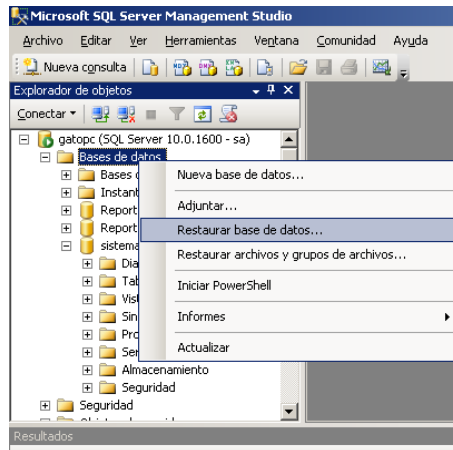


Figura 50 Opción restaurar base de datos

Nos muestra la siguiente imagen

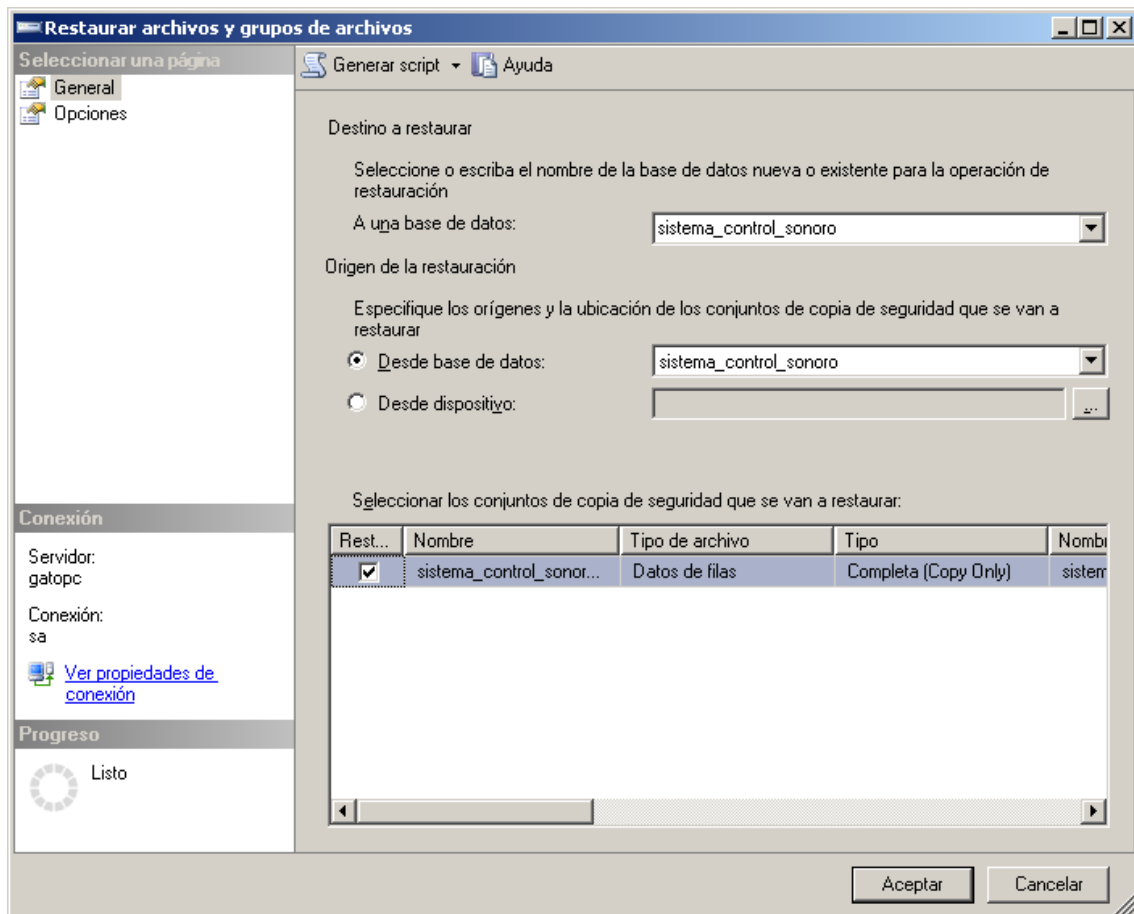


Figura 51 Selección de backup a restaurar

Escogemos nuestro backup y damos clic en aceptar. Y la base se ha restaurado, con lo cual nos damos cuenta de que la información es mantenida, de manera muy sencilla.

ANEXOS

5. Anexos, conclusiones y recomendaciones

I. Discusión.

- En el desarrollo de este trabajo de investigación se ha estudiado una nueva herramienta de manejo de datos a través de Visual Studio .NET. Es fundamental que se continúe con este trabajo, ya que permitirá podría ayudar de gran manera a la ciudad de Cuenca para detener la contaminación auditiva.
- El manejo de las herramientas del sistema integrado de desarrollo facilita y mejora el tiempo de desarrollo de aplicaciones empresariales.
- De la misma forma, al usar estas herramientas tecnológicas que no están maduras implica varias complicaciones ya sea en el aprendizaje como el futuro de la mismas.

II. Conclusiones.

Inicialmente se plantearon objetivos para la realización del sistema de control sonoro. Sobre los cuales se evaluarán las conclusiones.

El principal objetivo del proyecto se trataba de evitar, controlar el uso inadecuado de las bocinas en las unidades de transporte público, Aun no existe forma de evaluación dicha acción ya que al tratarse de un prototipo no se ha implementado el sistema en ninguna unidad de transporte público, no obstante, debido a que el sistema contiene un número máximo de pulsaciones para las bocinas, con lo cual si se puede hablar de un control con el solo hecho de instalar el dispositivo en la UDT.

Parte de la de información está incluida en el sistema ya que al evaluar el uso de la bocina de cada UDT, se emite un mensaje de texto al chofer de la unidad de transporte con lo cual éste se entera del mal uso de la bocina que le ha dado a su UDT.

Por otro lado el manejar información sobre las rutas nos brinda una información extra acerca de las rutas más conflictivas con respecto al tráfico vehicular y sobre el uso de las bocinas, sin embargo este parámetro de medición tendrá información valida si el sistema se implemente y pone en marcha en la UDT.

Finalmente el desarrollo de este sistema me ha permitido adquirir conocimientos tanto en las nuevas herramientas para el manejo de datos como el lenguaje de programación Visual Basic.NET así como el entorno de desarrollo Visual Studio. Profesionalmente el desarrollo del Sistema de control sonoro me ha permitido poner a prueba los conocimientos adquiridos a lo largo de estudio de la carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana.

A más de poner a prueba mis conocimientos el proyecto me ha ayudado a crecer profesionalmente incrementando mi incentivo por la investigación y la autoeducación

III. Recomendaciones.

El sistema de control sonoro únicamente se trata de un sistema que pretende controlar, el uso inadecuado de las bocinas, por lo cual se recomienda hacer campañas de concientización ya que el principal problema para la contaminación auditiva no es más que un particular cultural, por ende se puede remediar con mucha información y campañas de sanciones.

Con lo cual el sistema necesita trabajar mancomunadamente con las ordenanzas municipales así como la Jefatura de tránsito, a más de una ley que regule el uso de las bocinas.

De esta manera el sistema se volverá más eficiente y efectivo siendo así una buena opción para evitar en gran parte la contaminación auditiva.

IV. Bibliografía.

Joseph Scmuller UML en 24 horas, Activex No experience required , Dynamic HTML Master the Essentials

EUROSYSTEMS. Servicios Informáticos y Consultoría. Año 2003.

FIRMA Digital de la República de Argentina. Infraestructura de Firma Digital de la República de Argentina. Año 2006.

GRABBE Orlin J. Illustrated Digital Signatures. Año 2001.

GUADIANA María Guadalupe. Evaluación de Lotus Notes. Enero 2000.

HILERA José, MARTINEZ José. Ingeniería Documental Asistida por Computador. Año 1998.

IJENA LEIVA Renato Javier. Comercio, Derecho, Firma y Documentos Digitales o Electrónicos. Año 2001.

KNECHT-THURMANN Stephanie. Microsoft Small Business Server 2003: A clear and concise Administrator's Reference and How to. Agosto 2005.

LARRONDO María. Object Oriented Análisis, Design Notes. 2001.

MICROSOFT CORPORATION. Guíarápida de Microsoft Small Business Server 2003

MICROSOFT CORPORATION. Microsoft Windows Small Business Server 2003. Año 2006.

MICROSOFT CORPORATION. Guía de características de Share Point Services.

Microsoft Corporation June 2006 ADO.NET .NET Framework Language-Integrated Query

JesusLopez Seguridad en MS Sql Server 2008

Luis Miguel Blanco PROGRAMACIÓN EN VISUAL BASIC .NET
GuillermoSom "el Guille" Curso de Iniciacion de Visual Basic.NET

Joel Martin Chuco Marrufo Programando con Visual Basic .NET
Enriquillo Mañón Ramírez CrystalReports y VB.NET.

Seagate Crystal Report Manual del usuario, Seagate Software, Inc. 915 Disc Drive Scotts Valley California, EE.UU. 95066.

<http://es.scribd.com/doc/12800630/Manual-de-Crystal-Reports>

V. Anexos.

Script de creación de la base de datos.

-- CREACION DE TABLAS

--1. TABLA COOPERATIVAS

CREATE TABLE COOPERATIVAS

```
(
  ID_COOPERATIVA INT NOT NULL IDENTITY(1,1) PRIMARY KEY ,
  RUC_COOPERATIVA VARCHAR(13) NOT NULL,
  NOMBRE_COOPERATIVA VARCHAR(50) NOT NULL,
  OBSERVACION_COOPERATIVA VARCHAR(100)
)
```

--2. TABLA EMAIL_COOPERATIVAS

CREATE TABLE EMAIL_COOPERATIVAS

```
(
  ID_EMAIL_COOPERATIVA INT NOT NULL IDENTITY(1,1) PRIMARY KEY,
  DIRECCION_EMAIL_COOPERATIVA VARCHAR(60) NOT NULL,
  TIPO_CUENTA_EMAIL_COOPERATIVA VARCHAR(20) NOT NULL,
  OBSERVACION_EMAIL_COOPERATIVA VARCHAR(100),
  ID_COOPERATIVA INT REFERENCES COOPERATIVAS(ID_COOPERATIVA) ON
  DELETE CASCADE ON UPDATE CASCADE
)
```

--3. TABLA TELEFONO_COOPERATIVAS

CREATE TABLE TELEFONO_COOPERATIVAS

```
(
  ID_TELEFONO_COOPERATIVA INT NOT NULL IDENTITY(1,1) PRIMARY KEY,
  TIPO_NUMERO_TELEFONO_COOPERATIVA VARCHAR(10) NOT NULL,
  NUMERO_TELEFONO_COOPERATIVA VARCHAR(10) NOT NULL,
  ID_COOPERATIVA INT REFERENCES COOPERATIVAS(ID_COOPERATIVA) ON
  DELETE CASCADE ON UPDATE CASCADE
)
```

--4. TABLA DIRECCIONES COOPERATIVAS

CREATE TABLE DIRECCIONES_COOPERATIVAS

```
(
  ID_DIRECCION INT NOT NULL IDENTITY(1,1) PRIMARY KEY ,
```

```
CALLE_PRINCIPAL_DIRECCION VARCHAR(50) NOT NULL,  
  CALLE_SECUNDARIA_DIRECCION VARCHAR(50),  
  NUMERACION_PREDIAL_DIRECCION VARCHAR(10),  
  OBSERVACION_DIRECCION VARCHAR(200) ,  
  ID_COOPERAIVA INT REFERENCES COOPERATIVAS(ID_COOPERATIVA) ON  
DELETE CASCADE ON UPDATE CASCADE  
)
```

--5. TABLA DE BUSES

```
CREATE TABLE BUSES  
(  
  ID_BUS INT NOT NULL IDENTITY(1,1) PRIMARY KEY,  
  MARCA_BUS VARCHAR(10) NOT NULL,  
  MODELO_BUS VARCHAR(50) NOT NULL,  
  PLACLA_BUS VARCHAR(50) NOT NULL,  
  NUMERO_BUS INT NOT NULL UNIQUE,  
  ID_COOPERATIVA INT REFERENCES COOPERATIVAS(ID_COOPERATIVA) ON  
DELETE CASCADE ON UPDATE CASCADE  
)
```

--6. TABLA RUTAS

```
CREATE TABLE RUTAS  
(  
  ID_RUTA INT NOT NULL IDENTITY(1,1) PRIMARY KEY,  
  NOMBRE_RUTA VARCHAR(50),  
  LINEA_RUTA INT,  
  NOMBRE_OPERADORA_RUTA VARCHAR(50),  
  CONDICIONES_FLOTA_RUTA VARCHAR(50),  
  ID_COOPERATIVA INT REFERENCES COOPERATIVAS(ID_COOPERATIVA) ON  
DELETE CASCADE ON UPDATE CASCADE  
)
```

--7. RECORRIDOS

```
CREATE TABLE RECORRIDOS  
(  
  ID_RECORRIDOS INT NOT NULL IDENTITY(1,1) PRIMARY KEY,  
  NOMBRE_CALLE_IDA VARCHAR(50),  
  NOMBRE_CALLE_RETORNO VARCHAR(50),  
  NOMBRE_OPERADORA_RUTA VARCHAR(50),
```

```
ID_RUTA INT REFERENCES RUTAS(ID_RUTA) ON DELETE CASCADE ON
UPDATE CASCADE
)
```

--8. TABLA REPRESENTANTES

```
CREATE TABLE REPRESENTANTES
(
  ID_REPRESENTANTE INT NOT NULL IDENTITY(1,1) PRIMARY KEY,
  CEDULA_REPRESENTANTE VARCHAR(10) NOT NULL,
  NOMBRE_REPRESENTANTE VARCHAR(50) NOT NULL,
  DIRECCION_REPRESENTANTE VARCHAR(50),
  TELEFONO1_REPRESENTANTE VARCHAR(10),
  TELEFONO2_REPRESENTANTE VARCHAR(10),
  EMAIL_CHOFER VARCHAR(50),
  ID_COOPERATIVA INT REFERENCES COOPERATIVAS(ID_COOPERATIVA) ON
DELETE CASCADE ON UPDATE CASCADE
)
```

--9. TABLA DE CHOFERES

```
CREATE TABLE CHOFERES
(
  ID_CHOFER INT NOT NULL IDENTITY(1,1) PRIMARY KEY,
  CEDULA_CHOFER VARCHAR(10),
  NOMBRE_CHOFER VARCHAR(50),
  DIRECCION_CHOFER VARCHAR(50),
  TELEFONO1_CHOFER VARCHAR(10),
  TELEFONO2_CHOFER VARCHAR(10),
  EMAIL_CHOFER VARCHAR(50),
  ID_BUS INT REFERENCES BUSES(ID_BUS) ON DELETE CASCADE ON UPDATE
CASCADE
)
```

--10. TABLA DE LICENCIAS

```
CREATE TABLE LICENCIAS
(
  ID_LICENCIA INT NOT NULL IDENTITY(1,1) PRIMARY KEY,
  TIPO_LICENCIA VARCHAR(10) NOT NULL,
  TIPO_SANGRE VARCHAR(10) NOT NULL,
```

```
FECHA_EMISION_LICENCIA DATE NOT NULL,  
FECHA_CADUCIDAD_LICENCIA DATE NOT NULL,  
ID_CHOFER INT REFERENCES CHOFERES(ID_CHOFER) ON DELETE CASCADE  
ON UPDATE CASCADE  
)
```

```
--11. TABLA DE USUARIOS  
CREATE TABLE USUARIOS
```

```
(  
  ID_USUARIO INT NOT NULL IDENTITY(1,1) PRIMARY KEY,  
  NOMBRE_USUARIO VARCHAR(40) NOT NULL,  
  CLAVE_USUARIO VARCHAR(40) NOT NULL,  
  CEDULA_USUARIO VARCHAR(10),  
  NOMBRES_USUARIO VARCHAR(50),  
  TIPO_USUARIO VARCHAR(1) NOT NULL,  
  OBSERVACION_USUARIO VARCHAR(100)  
)
```

```
--12. TABLA DE DISPOSITIVO  
CREATE TABLE DISPOSITIVOS
```

```
(  
  ID_DISPOSITIVO INT NOT NULL IDENTITY(1,1) PRIMARY KEY,  
  SERIAL_DISPOSITIVO VARCHAR(4) NOT NULL,  
  FECHA_INGRESO DATE NOT NULL,  
  OBSERVACION_DISPOSITIVO VARCHAR(10)  
)
```

```
--13. TABLA DE USO_BOCINAS  
CREATE TABLE USO_BOCINAS
```

```
(  
  ID_USO_BOCINA INT NOT NULL IDENTITY (1,1) PRIMARY KEY,  
  HORA_USO_BOCINA VARCHAR(20),  
  DIA_SEMANA VARCHAR(20) NOT NULL,  
  FECHA_USO_BOCINA DATE,  
  DECIBEL_ALTO_USO_BOCINA NUMERIC(10,10) NOT NULL,  
  DECIBEL_BAJO_USO_BOCINA NUMERIC(10,10) NOT NULL,  
  DECIBEL_MEDIO_USO_BOCINA NUMERIC(10,10) NOT NULL,  
  DURACION_MILISEGUNDOS_USO_BOCINA NUMERIC(10,10) NOT NULL,  
)
```

```
ID_BUS INT REFERENCES BUSES(ID_BUS) ON DELETE CASCADE ON UPDATE  
CASCADE,  
ID_DISPOSITIVO INT REFERENCES DISPOSITIVOS(ID_DISPOSITIVO) ON DELETE  
CASCADE ON UPDATE CASCADE,  
  ID_USUARIO INT REFERENCES USUARIOS(ID_USUARIO) ON DELETE  
CASCADE ON UPDATE CASCADE  
)
```