INSTITUTO TECNOLOGICO
"CORDILLERA"

CARRERA DE ANÁLISIS DE SISTEMAS

MEJORAMIENTO DEL PROCESO DE FACTURACIÓN MEDIANTE UNA APLICACIÓN WEB DE FACTURACIÓN ELECTRÓNICA PARA EL CENTRO DESARROLLO INFANTIL "DESCUBRIR" DEL DISTRITO METROPOLITANO DE QUITO

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

Autor: Mosquera Martínez Marco Renato

Tutor: Lic. Patricia Garzón

Quito, Abril 2015

DECLARATORIA

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

_____

Marco Renato Mosquera Martínez

CC 171626844-4

CESIÓN DE DERECHOS

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

PRIMERA: ANTECEDENTE.- a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado "Mejoramiento del proceso de facturación mediante una aplicación web de facturación electrónica

para el Centro Desarrollo Infantil "Descubrir" del distrito metropolitano de Quito, el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. **b)** Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

SEGUNDA: CESIÓN Y TRANSFERENCIA.- Con el antecedente indicado, el

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

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

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

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

SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.- Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de

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

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

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

En aceptación firman a los diez días del mes de abril del dos mil quince.

f) _____
C.C. Nº 1716268444

CEDENTE

f) _____
Instituto Superior Tecnológico Cordillera

CESIONARIO

CESIÓN DE DERECHOS

Yo, Marco Renato Mosquera Martínez alumno de la Escuela de Sistemas, libre y voluntariamente cedo los derechos de autor de mi investigación en favor Instituto Tecnológico Superior "Cordillera".

_____

CC 171626844-4

AGRADECIMIENTO

Doy un sentido agradecimiento por el apoyo incondicional que me otorgaron mis maestros en todo el proceso educativo realizado en esta prestigiosa institución educativa y de forma especial a mi maestra y tutora Lcda. Patricia Garzón que colaboro en la guía para el desarrollo del proyecto de grado.

Gracias también a todos mis familiares, a mis hijas y esposa que son el pilar que motiva cada uno de mis esfuerzos para la culminación del proyecto.

DEDICATORIA

A mis padres por su apoyo incondicional
A mi esposa por su amor sin condiciones
A mis hermanos por su apoyo constante
A mis hijas por infinito amor

INDICE

Titulo	Página
Capítulo I: Antecedentes	1
1.01 Contexto	1
1.02 Justificación	2
1.03 Definición del Problema Central	4
Capítulo II: Análisis de Involucrados	6
2.01 Requerimientos	6
2.01.01 Descripción del sistema actual.	6
2.01.02 Visión y alcance.	7
2.01.03 Entrevista.	9
2.01.04 Matriz de requerimientos.	9
2.01.05 Descripción detallada.	10
2.02 Mapeo de Involucrados	18
2.03 Matriz de Involucrados	19
Capítulo III: Problemas y Objetivos	20
3.01 Árbol de Problemas	20
3.02 Árbol de Objetivos	21
3.03 Diagrama de Casos de Uso	22
3.04 Especificación de Casos de Uso	23
3.05 Casos de Uso de Realización	24
3.06 Especificación de Casos de uso de realización	25
3.07 Diagramas de secuencia del sistema	26
Capítulo IV: Análisis de Alternativas	27
4.01 Matriz de Análisis de Alternativas	27
4.02 Matriz de Impacto de Objetivos	29
4.03 Estándares para el Diseño de Clases	30
4.03.01 Clases	30
4.03.02 Atributos y Métodos:	30
4.03.03 Métodos	31
4.04 Diagrama de Clases	32
4.05 Modelo Lógico y Físico	33
4.05.01 Modelo Lógico	33

4.05.01 Modelo Físico 34
4.06 Diagrama de Componentes 35
4.07 Diagrama de Estrategias 35
4.08 Matriz de Marco Lógico 36
4.09 Vistas Arquitectónicas 37
4.09.01 Vista Lógica 37
4.09.02 Vista Física 38
4.09.03 Vista de Desarrollo 38
4.09.04 Vista de Procesos 39
Capítulo V: Análisis de Alternativas 41
5.01 Especificación de estándares de programación 41
5.01.01 Declaraciones de Variables 42
5.01.02 Descripción de la variable 42
5.01.03 Variables de Tipo Arreglo 43
5.01.04 Definición de Controles 43
5.02 Diseño de Interfaces de Usuario 47
5.03 Especificación de pruebas de unidad 49
5.04 Especificación de pruebas de aceptación 52
5.05 Especificación de pruebas de carga 54
5.06 Configuración del Ambiente mínima/ideal 57
Capítulo VI Aspectos Administrativos 59
6.01 Recursos 59
6.02 Presupuesto 60
6.03 Cronograma de Actividades 61
Capítulo VII: Conclusiones y Recomendaciones 62
7.01 Conclusiones 62
7.02 Recomendaciones 63

INDICE DE TABLAS

Titulo **Página**

Tabla 1 *Matriz de Análisis de Fuerza T* 4

Tabla 2 *Entrevista* 65

Tabla 3 *Detalle de requerimientos* 67

Tabla 4 *Detalle de requerimiento Funcional RF001* 10

Tabla 5 *Detalle de requerimiento Funcional RF002* 11

Tabla 6 *Detalle de requerimiento Funcional RF003* 12

Tabla 7 *Detalle de requerimiento Funcional RF004* 13

Tabla 8 *Detalle de requerimiento Funcional RF005* 14

Tabla 9 *Detalle de requerimiento no Funcional RNF001* 15

Tabla 10 *Detalle de requerimiento no Funcional RNF002* 16

Tabla 11 *Detalle de requerimiento no Funcional RNF003* 17

Tabla 12 *Detalle interés, Involucrados y conflictos percibidos del proyecto* 69

Tabla 13 *Especificación Caso de Uso de Contexto (General) CU001* 23

Tabla 14 *Especificación Caso de Uso de Realización CUR001* 25

Tabla 15 *Análisis estadístico de alternativas* 27

Tabla 16 *Detalle de valores del análisis de alternativas* 28

Tabla 17 *Análisis de los objetivos* 29

Tabla 18 *Resumen del proyecto que destaca lo que se desea lograr* 36

Tabla 19 *Detalle de la descripción de un variable* 42

Tabla 20 *Detalle del tipo de datos de una variable* 44

Tabla 21 *Detalle del nombre descriptivo para controles* 45

Tabla 22 *Detalle de como declarar variables, atributos y objetos* 45

Tabla 23 *Detalle de como declarar clases* 46

Tabla 24 *Detalle de como declarar Métodos* 46

Tabla 25 *Detalle de como declarar Funciones* 47

Tabla 26 *Prueba de interface de usuario (estándares)* 50

Tabla 27 *Pruebas de Reportes, resultados eficientes* 50

Tabla 28 *Pruebas de compilación de Código* 51

Tabla 29 *Pruebas de Almacenamiento de datos en la Base* 51

Tabla 30 *Detalle de pruebas de aceptación en la creación de usuarios* 52

Tabla 31 *Detalle de pruebas de aceptación en el proceso de Institución* 52
Tabla 32 *Detalle de pruebas de aceptación en el proceso de Alumno* 53
Tabla 33 *Detalle de pruebas de aceptación en el proceso de Representante* 53
Tabla 34 *Detalle de pruebas de aceptación en el proceso de Facturación Electrónica* 54
Tabla 35 *Detalle de un tipo de prueba de carga más baja* 55
Tabla 36 *Detalle de un tipo de prueba de carga con un número mínimo de usuarios* 56
Tabla 37 *Detalle de un tipo de prueba de carga con un número máximo de usuarios* 56
Tabla 38 *Detalle de un tipo de prueba de carga con un número máximo de usuarios ejecutándose hasta llegar al colapso. Se requiere descubrir los límites* 57

INDICE DE FIGURAS

Titulo **Página**

Figura 1 Detalle de los involucrados con sus relaciones 19
Figura 2 Organización de Causales, Problema Central y Efectos 20
Figura 3 Detalle de objetivos para llegar a una conclusión y finalidad 21
Figura 4 Caso de uso de Contexto (General) CU001 22
Figura 5 Caso de uso de realización General CUR001 24
Figura 6 Diagrama de Secuencia General 26
Figura 7 Donde se especifica los atributos y acciones del sistema 32
Figura 8 Donde se agrega la persistencia de cada clase 33
Figura 9 Donde se genera las tablas en base a los diagramas anteriores 34
Figura 10 Donde se especifica los módulos y las capas del sistema 35
Figura 11 Donde se especifica las estrategias para llegar a una finalidad 35
Figura 12 Descripción de la lógica del sistema 37
Figura 13 Descripción física del sistema 38
Figura 14 Descripción detallada del sistema mediante componentes 38

Figura 15 Vista del Proceso Ingreso de Datos 39
Figura 16 Vista del Proceso Facturación 40
Figura 17 Diagrama de secuencia del proceso de pruebas de aceptación 42
Figura 18 Capas en donde interactúa la interface del usuario 48
Figura 19 Diagrama general de los pasos para diseñar una interface 48
Figura 20 Diseño general de interface de usuario 49
Figura 21 Tiempos propuestos para el desarrollo y la implementación del Sistema 61

RESUMEN EJECUTIVO

El proyecto realizado tiene como fin mejorar el proceso de facturación del Centro de Desarrollo infantil Descubrir de la ciudad de quito.

Las distintas actividades que se realizaron para el óptimo aprovechamiento de la información otorgada por el CDI fueron; análisis, desarrollo, pruebas, implementación de la aplicación desarrollada con metodología UML que permite tener un control claro de los procesos realizados para la elaboración de sistemas informáticos.

La utilización de arquitectura Web permite trabajar en capas como son: Capa de lógicas del negocio, Capa de Presentación y Acceso a datos.

La presente investigación se encuentra estructurada por capítulos: Capitulo I El problema, 1.01 Planteamiento del problema, 1.02 Formulación del problema, 1.03 Objetivo General, 1.04 Objetivos Específicos; Capitulo II Análisis de involucrados, 2.01 Requerimientos 2.01.01 Descripción del sistema actual, 2.01.02 Visión y Alcance, 2.01.03 Entrevistas 2.01.04Matriz de requerimientos, 2.01.05 Descripción detallada, 2.01.06 Mapeo de Involucrados, 2.01.07 Matriz de Involucrados; Capitulo III Problemas y Objetivos, 3.01 Árbol de Problemas, 3.02 Árbol de Objetivos, 3.03 Diagrama de Casos de Uso 3.04, Especificación de Casos de Uso, 3.05 Casos de Uso de Realización 3.06, Especificación de Casos de uso de realización 3.07 Diagramas de secuencia del sistema; Capitulo IV Análisis de alternativas, 4.01 Matriz de Impacto de Objetivos, 4.02 Matriz de Análisis de Alternativas, 4.03 Estándares para el Diseño de Clases 4.03.01 Clases, 4.03.02 Atributos y Métodos, 4.03.03 Métodos, 4.04 Diagrama de Clases, 4.05 Modelo Lógico y Físico, 4.05.01 Modelo Lógico, 4.05.02 Modelo Físico; 4.06 Diagrama de Componentes, 4.07 Diagrama de Estrategias, 4.08 Matriz de Marco Lógico, 4.09 Vistas

Arquitectónicas, 4.09.01 Vista Lógica, 4.09.02 Vista Física, 4.09.03 Vista de Desarrollo, 4.09.04 Vista de Procesos; Capítulo V Propuesta, 5.01 Especificación de estándares de Programación, 5.01.01 Declaración de Variables, 5.01.02 Descripción de variables, 5.01.03 Variables de tipo Arreglo, 5.01.04 Definición de Controles, 5.02 Diseño de interface de Usuario, 5.03 Especificación de Pruebas de Unidad, 5. 04 Especificación de pruebas de Aceptación, 5.05 Especificación de Pruebas de Carga, 5.06 Configuración del Ambiente mínimo/Ideal; Capítulo VI Aspectos Administrativos, 6.01 Recursos, 6.02 Presupuesto, 6.03 Cronograma; Capitulo VII Conclusiones y recomendaciones, 7.01 Conclusiones, 7.02 Recomendaciones.

ABSTRACT

This project involves the analysis, design, development and implementation of a computer system to support the academic management of the Republic of Venezuela School. The purpose of this platform is to automate the management and care of school processes.

The project conducted aims to improve the billing process Child Development Center Discover the city of Quito.

The various activities conducted for the optimal use of information provided by the CDI were; analysis, development, testing, implementation of the application developed with UML methodology to have a clear control of the processes carried out for the development of computer systems.

The use of Web architecture allows working in layers such as: Business Logic Layer, Presentation Layer and Data Access.

This research is structured into chapters: Chapter I The problem, Problem 1.01, 1.02 Formulation of the problem, General Purpose 1.03, 1.04 Specific Objectives; Chapter II Analysis involved, 2.01 Requirements 2.01.01 Description of the current system, Vision and Scope 2.01.02, 2.01.03 2.01.04 Interviews Matrix requirements detailed description 2.01.05, 2.01.06 Mapping Involved, 2.01.07 Involved matrix; Chapter III Problems and objectives, 3.01 Problem Tree, Objective Tree 3.02, 3.03 Use Case Diagram 3.04 Use Case Specification, Use Cases 3.05 3.06 Carrying, Case Specification 3.07 embodiment using sequence diagrams system; Chapter IV Alternatives Analysis, Impact Matrix 4.01 Lenses 4.02 Alternatives Analysis Matrix, 4.03 Design Standards 4.03.01 Classes, Attributes and Methods 4.03.02, 4.03.03 Methods, 4.04 Class Diagram, 4.05 Logical and Physical Model, Logical Model 4.05.01, 4.05.02 Physical Model; 4.06 Diagram Components, Strategies Diagram

4.07, 4.08 Logical Framework Matrix, 4.09 Views Architectural, Vista Logic 04/09/01, 04/09/02 Physical Vista, Vista Development 04/09/03, 04/09/04 View Process; Chapter V Motion, 5.01 specification standards Programming, 5.01.01 Declaration of Variables, 5.01.02 Description of variables Variable type 5.01.03 Arrangement, 05.01.04 Definition of controls, 5.02 User Interface Design, Specification 5.03 Unit Testing, 5. 04 test specification Acceptance Test Specification 5.05 Load Settings minimum 5.06 / Ideal Environment; Chapter VI Administrative Aspects, 6.01 Resources, Budget 6.02, 6.03 Schedule; Chapter VII Conclusions and recommendations Conclusions 7.01, 7.02 Recommendations

Capítulo I: Antecedentes

1.01 Contexto

La factura electrónica es un documento digital que tiene la misma validez que una factura impresa, esta factura electrónica se puede intercambiar por medios electrónicos pero siempre debe contar con una firma digital para que tenga el mismo aval que la factura física.

(«FACTURACIÓN ELECTRÓNICA - QUÉ ES FACTURACIÓN ELECTRÓNICA | software Facturación electrónica Ecuador - NECTILUS - Soluciones agiles, somos eficaces y eficientes en desarrollar software», s. f.-a)

La facturación electrónica en el Ecuador es un proceso que se lo viene insertando desde el año 2014 en el país ya que es parte de la nueva normativa que maneja el Servicio de Rentas Internas (SRI) y se aplica a todo tipo de organizaciones privadas, públicas o personas naturales que están obligadas a llevar contabilidad al realizar estos cambios se beneficiaran principalmente las empresas, ya que este proceso se lo realiza con mayor eficiencia reduciendo así los costos generados por la factura física, agilitando los procesos y asegurando la integridad de la información y teniendo un enlace directo entre la institución que genera la factura electrónica y el Servicio de Rentas Internas ya que el SRI cuentan con un respaldo magnético de la factura, los clientes que reciben a su correo electrónico la factura electrónica y la descarga e imprime el representante de su correo electrónico de esta manera se logra un equilibrio entre todos los involucrados en este proceso.

(«Factura Electronica | software Facturación electrónica Ecuador - NECTILUS - Soluciones agiles, somos eficaces y eficientes en desarrollar software», s. f.)

A nivel nacional son pocas instituciones educativas del país que cuentan con sistemas informáticos en general para ayudar a la gestión administrativa y muchas menos en la gestión contable debido también a la falta de interés de las instituciones educativas para mejorar sus condiciones de trabajo y así agilitar sus procesos de forma efectiva.

Dado que la ley expresa la obligatoriedad de entregar facturas electrónicas en todas las actividades empresariales, educativas, comerciales se genera la necesidad de las instituciones educativas de la ciudad y de todo el país de contar con programas o paquetes informáticos que agilite la emisión de facturación electrónica para mejorar el desempeño laboral del personal encargado de este proceso y cumplir con la normativa vigente.

1.02 Justificación

Al conocer cuál es la situación del Centro de Desarrollo Infantil Descubrir (CDI) al realizar el proceso de facturación se puede manejar varias alternativas para mejorar la forma de trabajar del personal de la institución y agilitar este proceso.

Contratar una empresa de desarrollo software informático que se especialice en sistemas de facturación para que realice un sistema de facturación web electrónica tendría un costo elevado y no se tendría la seguridad que la aplicación que elabore la empresa de desarrollo cuente con los requerimientos necesarios del CDI.

El comprar un paquete informático genérico que está destinado a realizar el proceso de facturación de forma general para cualquier tipo de empresa es decir que no está diseñado específicamente para realizar la facturación electrónica de un CDI puede resultar muy costoso ya que por el hecho de usarcé en diversas empresas o

instituciones su paquete informático es muy extenso y tendrían que utilizar solo cierta parte de toda la aplicación es decir al módulo que necesita el CDI, la institución educativa tendría que adaptarse al sistema y esto ocasionaría problemas ya que el personal es renuente a los cambios y tardaría mucho tiempo en adaptarse a este cambio.

El desarrollo de una aplicación de web de facturación electrónica que cuente con todos los requerimientos de la institución educativa en este caso del CDI, involucrándonos con su necesidad de mejorar este proceso de facturación, realizando el levantamiento de la información directamente con los involucrados por medio de una entrevista y la observación de cómo se realiza el proceso de facturación en la actualidad permitirá tener una visión clara de los inconvenientes que tiene el proceso de facturación y buscar la forma más práctica y sencilla de trabajar y mejorar su tiempo de respuesta, asegurar la integridad de la información, que las facturas emitidas de forma electrónica cuente con toda la información que exige el SRI y económicamente tendrán un gran beneficio ya que el costo de la aplicación será mínimo para el CDI.

1.03 Definición del Problema Central

Tabla 1

Matriz de Análisis de Fuerza T

Situación Empeorada	Situación Actual				Situación Mejorada
Demora en los proceso de registro de la información en la factura e inconsistencia de la información entregada al SRI	Inadecuado manejo del proceso de facturación.				Eficiencia y seguridad en el manejo de la información de las facturas de cada estudiante.
Fuerzas Impulsadoras	**I**	**PC**	**I**	**PC**	**Fuerzas Bloqueadoras**
Capacitar a la persona involucrada con el proceso de facturación para que su trabajo sea óptimo.	3	5	5	3	Poco interés del personal por mejorar el proceso de facturación.
Mantener los datos de las facturas de cada estudiante actualizado y consistente, la información de las facturas disponible a cada momento.	4	5	5	4	Duplicidad de documentos e información de los pagos dispersa
Agilitar la respuesta al momento de requerir de listado de los pagos de los estudiantes.	3	5	5	3	Demora en la búsqueda debido a la gran cantidad de información de cada alumno.
Mantener la información de las facturas segura e integra.	4	5	5	4	Alto riesgo de inconsistencia de la información de las facturas.
Contar con una aplicación de facturación electrónica que agilite él proceso.	1	5	5	1	Ausencia de aplicaciones informáticas que agilicen el proceso de facturación.

Realizando un análisis de la matriz de fuerzas totales podemos identificar cual es la situación actual del Centro de Desarrollo Infantil Descubrir que muestra que el proceso de facturación en la actualidad no es el más adecuado ya que es proceso no se lo realiza de manera óptima.

Lo que podría derivar en una situación desmejorada ya que el trato de la información de las facturas al llenar los datos de los clientes y los valores a cancelar por los padres de familia se encuentra vulnerables e inconsistentes y que al incrementar el número de estudiantes y por ende de pagos de sus pensiones existirá demora en el proceso del entrega de las facturas. Se puede mejorar de gran manera capacitando al personal que interviene en este proceso, manteniendo los datos de la parte de facturación actualizada, agilitando los tiempos al momento de entregar la información de los pagos, manteniendo la información segura y consistente en archivos magnéticos y teniendo una aplicación que permita obtener la información cuando se la necesite.

Teniendo en cuenta todas las fuerzas impulsadoras se puede llegar a mejorar todo el proceso de facturación electrónica hasta lograr eficiencia del personal encargado, seguridad en el proceso y que se cumpla con lo que exige la ley.

Capítulo II: Análisis de Involucrados

2.01 Requerimientos

2.01.01 Descripción del sistema actual.

En el Centro de Desarrollo Infantil Descubrir el proceso de facturación se realiza de forma manual ya que no se utiliza una aplicación informática que facilite este trabajo, el registro de los pagos.

Al momento que los clientes que son los o padres de familia de los estudiantes del CDI realizan el pago en efectivo concerniente a matriculas la secretaria entrega una factura física, luego el pago de las pensiones de los niños los cuales su valor depende del horario que asisten los niños al CDI, el horario se divide en: media jornada hasta la 1:00 pm, media jornada con almuerzo hasta 14:30 y jornada completa hasta 17:00. Y este pago se lo realiza los hasta los 10 primeros días de cada mes después de lo cual se envía una notificación al padre de familia informándole que aún no realiza su pago mensual.

Después de haber realizado el pago en efectivo la señora secretaria procede a llenar la factura que consta con cabecera con la respectiva autorización del SRI, datos de informativos del CDI y donde se escribe datos como la fecha que se realiza el pago, nombre del cliente, dirección del cliente, teléfono del cliente y su número de identificación.

Se llena el detalle de la factura que consta de Cantidad, descripción que se realiza el pago, el valor unitario, el valor total, total en dólares.

Si la información de la factura es correcta la secretaria o la directora firman en el espacio asignado para ellas y el cliente firma en el espacio de recibí conforme. Después se entrega el documento original al representante del estudiante y dos copias guarda el CDI como constancia del pago.

Las copias de los comprobantes de los pagos se guarda en archivadores y se lleva un registro en un libro diario de las fechas de los pagos y de los montos recibidos de cada estudiante y el número de factura, también se registra aquí las personas que aún no realizan los pagos y los montos que adeudan al CDI, con esta manera de llevar el proceso de facturación la representante del CDI realiza las declaraciones al SRI, las mismas que suelen presentar inconsistencias.

2.01.02 Visión y alcance.

La finalidad del desarrollo de este sistema web de facturación electrónica se basa en mejorar este proceso de facturación del CDI.

Esta investigación permite conocer todo el manejo del proceso de facturación del Centro de Desarrollo Infantil Descubrir y saber que una aplicación web de facturación electrónica mejorará el tiempo de ejecución y respuesta dando así una solución integral a la problemática en la cual se encuentra inmerso el CDI al receptar los pagos de matrícula y pensiones, ya que la misma contara con orden adecuado, la seguridad e integridad de la información y lo más importante que es cumplir con la normativa vigente del SRI.

El registro de los pagos de matrícula y pensiones de cada año lectivo se lo realizara dentro del sistema de facturación web electrónica el mismo que genera una factura electrónica que cuenta con una firma digital electrónica autorizada por el Servicio de

Rentas Internas y que es emitida por Banco Central del Ecuador, se enviará la factura al correo electrónico entregado por el representante de los estudiantes.

Los reportes de los pagos se generan, por alumnos, alumnos impagos y por mes que permita saber cuáles alumnos faltan por pagar ya que esta es la información primaria es la que necesita la directora y la secretaria para tener un control adecuado de este proceso.

Se entregara un manual técnico en el cual se detalla cómo está constituida la aplicación es decir los programas utilizados para el desarrollo de la aplicación web.

Manual de usuario el cual permitirá conocer el funcionamiento a fondo de toda la aplicación de facturación electrónica, como ingresar los datos de la institución, del representante, como se realiza en ingreso de los usuarios con sus respectivos perfil, momo realizar el mantenimiento de cada uno de las tablas, como generar la factura, como enviar la factura a cada representante, etc.

Manual de instalación que indica los requisitos mínimos que debe tener el computador para que la aplicación funcione de manera correcta es decir que software y hardware.

2.01.03 Entrevista.

Con la entrevista podremos conocer la problemática de la institución y realizar un análisis que permita saber cómo mejorar esa situación. (Ver anexo A.01)

2.01.04 Matriz de requerimientos.

Aquí es donde se detalla los requerimientos separando los funcionales de los no funcionales esto permitirá tener un conocimiento claro de que necesita la institución para su aprovechamiento total. (Ver anexo A.02)

2.01.05 Descripción detallada.

Tabla 4

Detalle de requerimiento Funcional RF001

La Directora y o el administrador del sistema de facturación del CDI necesitan llenar el formulario de registro del alumnos y representantes.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RF001**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**Funcional**
Datos de Entrada	Cédula del Administrador del sistema		
Descripción	El usuario (Directora) debe ingresar al sistema en el cual podrá ingresar todos los datos del alumno y sus representantes, necesarios para la generación de la factura electrónica.		
Datos de salida	Datos del estudiante Datos del representante		
Resultados Esperados	Los resultados esperados con este requerimiento es dar la facilidad que usuario (Directora) ingrese los datos de los estudiantes y representantes de forma sencilla.		
Origen	Dirección		
Dirigido a	Usuarios (Directora)		
Prioridad	5		
Requerimientos Asociados	RNF002		
ESPECIFICACIÓN			
Precondiciones	1. El usuario es Administrador (Directora). 2. El usuario debe estar registrado en el sistema RNF002.		
Poscondiciones	1. Si el usuario (Directora) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2. Si los datos ingresados son correctos podrá realizar la factura electrónica RF004.		
Criterios de Aceptación	Permite que un usuario (Directora) realizar el ingreso de la información del estudiante y representante al sistema RF001.		

Tabla 5

Detalle de requerimiento Funcional RF002

La Directora y o el administrador del sistema del CDI necesitan llenar el formulario del representante financiero si los estudiantes son apadrinados.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RF002**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**Funcional**
Datos de Entrada	Cédula del Administrador del sistema		
Descripción	El usuario (Directora) debe ingresar al sistema en el cual podrá ingresar todos los datos del representante financiero de los alumnos que tienen auspicio de empresas en las que trabajan sus padres.		
Datos de salida	Datos del representante financiero.		
Resultados Esperados	Los resultados esperados con este requerimiento es dar la facilidad que usuario (Directora) ingrese los datos del representantes financiero de forma sencilla.		
Origen	Dirección		
Dirigido a	Usuarios (Directora)		
Prioridad	4		
Requerimientos Asociados	RF001		
ESPECIFICACIÓN			
Precondiciones	1. Para ejecutar el requerimiento primero el alumno debe estar registrado en el sistema RF001. 2. El usuario es Administrador (Directora). 3. El usuario debe estar registrado en el sistema RNF002.		
Poscondiciones	1. Si el usuario (Directora) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2. Si los datos ingresados son correctos podrá realizar la factura electrónica RF004.		
Criterios de Aceptación	Permite que un usuario (Directora) realizar el ingreso de la información del representante financiero al sistema.		

Tabla 6

Detalle de requerimiento Funcional RF003

El administrador del sistema del CDI necesita llenar el formulario de información de la institución.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RF003**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**Funcional**
Datos de Entrada	Cédula del Administrador del sistema		
Descripción	El usuario (Directora) debe ingresar al sistema en el cual podrá ingresar todos los datos de la institución, necesaria para la generación de la factura electrónica.		
Datos de salida	Datos de la Institución.		
Resultados Esperados	Los resultados esperados con este requerimiento es dar la facilidad que usuario (Directora) ingrese los datos de la institución que constaran en la cabecera de la factura.		
Origen	Dirección		
Dirigido a	Usuarios (Directora)		
Prioridad	5		
Requerimientos Asociados	RF004		
ESPECIFICACIÓN			
Precondiciones	1. El usuario es Administrador (Directora). 2. El usuario debe estar registrado en el sistema RNF002.		
Poscondiciones	1. Si el usuario (Directora) no se acuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2. Si los datos ingresados son correctos podrá realizar la factura electrónica RF004.		
Criterios de Aceptación	Permite que un usuario (Directora) realizar el ingreso de la información de la institución al sistema.		

Tabla 7

Detalle de requerimiento Funcional RF004

Generar factura electrónica con los datos de institución, representante financiero y de alumno.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RF004**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**Funcional**
Datos de Entrada	Cédula del usuario (Directora o Secretaria)		
Descripción	El usuario (Directora o Secretaria) debe ingresar al sistema y con los datos de la institución, estudiante, representante financiero y los rubros que debe cancelar se generara una factura electrónica.		
Datos de salida	Factura generada.		
Resultados Esperados	Los resultados esperados con este requerimiento generar las facturas.		
Origen	Dirección		
Dirigido a	Usuarios (Directora o Secretaria)		
Prioridad	5		
Requerimientos Asociados	RF001, RF002, RF003, RNF002		
ESPECIFICACIÓN			
Precondiciones	1. Para ejecutar el requerimiento primero deben estar los datos de institución RF003, representante financiero RF002 y alumno ingresados en el sistema RF001. 2. El usuario debe ser Directora o Secretaria. 3. El usuario debe estar registrado en el sistema RNF002.		
Poscondiciones	1. Si el usuario (Directora o Secretaria) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2. Si los datos son correctos se podrá guardar, firmar, enviar la factura electrónica RF005.		
Criterios de Aceptación	Permite que un usuario (Directora o Secretaria) generar las facturas electrónicas.		

Tabla 8

Detalle de requerimiento Funcional RF005

Firmar electrónicamente, guardar y enviar el archivo XML para que el SRI lo valide.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RF005**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**Funcional**
Datos de Entrada	Cédula del Administrador del sistema		
Descripción	El usuario (Directora) debe ingresar al sistema en el cual podrá firmar la factura electrónica, guardar y enviar al SRI para que valide la factura.		
Datos de salida	Archivo XML.		
Resultados Esperados	Los resultados esperados con este requerimiento son firmar guardar y enviar las facturas.		
Origen	Dirección		
Dirigido a	Usuarios (Directora)		
Prioridad	5		
Requerimientos Asociados	RF004		
ESPECIFICACIÓN			
Precondiciones	1. Para ejecutar el requerimiento primero debe estar generada la factura electrónica RF004. 2. El usuario es Administrador (Directora). 3. 3. El usuario debe estar registrado en el sistema RNF002.		
Poscondiciones	1. Si el usuario (Directora) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2. Si el usuario es administrador (Directora) podrá enviar la factura validada a los correos electrónicos de los representantes financieros.		
Criterios de Aceptación	Permite que un usuario (Directora) firmar, guardar y enviar las facturas electrónicas.		

Tabla 9

Detalle de requerimiento no Funcional RNF001

La Directora y la Secretaria quieren ingresar desde cualquier lugar a revisar los pagos realizados de matrícula o pensiones.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RNF001**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**No Funcional**
Datos de Entrada	Cédula del usuario (Directora o Secretaria)		
Descripción	El usuario (Directora o Secretaria) debe ingresar al sistema desde cualquier lugar y podrán revisar los pagos ingresados respecto a matriculas o pensiones.		
Datos de salida	Reporte de pagos de cada estudiante, por paralelo y por mes.		
Resultados Esperados	Los resultados esperados con este requerimiento es mejorar la manera de llevar información y a su vez pueda consultar y visualizar reportes de una manera eficiente.		
Origen	Dirección		
Dirigido a	Usuarios (Directora o Secretaria)		
Prioridad	5		
Requerimientos Asociados	RF005		
ESPECIFICACIÓN			
Precondiciones	1. Para ejecutar el requerimiento primero la factura ya debió ser generada 2. El usuario debe ser la Directora o Secretaria. 3. Una vez ingresado al sistema el usuario debe buscar Reportes.		
Poscondiciones	1. Si el usuario (Directora o Secretaria) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2. Si el usuario es administrador (Directora) podrá revisar los reportes.		
Criterios de Aceptación	Permite que un usuario (Directora o Secretaria) tener un manejo instantáneo de la información de las facturas.		

Tabla 10

Detalle de requerimiento no Funcional RNF002

Se requiere que las personas que tengan acceso al sistema de facturación sean un Administrador y Secretaria		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RNF002**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**No Funcional**
Datos de Entrada	Cédula del Administrador del sistema		
Descripción	El usuario (Directora) debe ingresar al sistema en el cual podrá ingresar los usuarios con su respectivo rol para que tenga acceso a ciertas opciones del sistema.		
Datos de salida	Usuarios ingresados.		
Resultados Esperados	Los resultados esperados con este requerimiento es dar la facilidad que los usuario tenga acceso a ciertas opciones del sistema.		
Origen	Dirección		
Dirigido a	Usuarios (Directora)		
Prioridad	4		
Requerimientos Asociados	Ninguno		
ESPECIFICACIÓN			
Precondiciones	1. Para ejecutar el requerimiento primero el usuario debe estar registrado en el sistema 2. El usuario debe ser la Directora. 3. Una vez ingresado al sistema el usuario debe ser administrador del sistema.		
Poscondiciones	1. Si el usuario (Directora) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave. 2. El usuario tendrá acceso a las distintas opciones del sistema.		
Criterios de Aceptación	Permite que un usuario (Directora) crear nuevos usuarios para el sistema.		

Tabla 11

Detalle de requerimiento no Funcional RNF003

La Directora o Secretaria necesitan enviar la factura electrónica ya validada a los correos electrónicos de los representantes financieros de los estudiantes.		**Estado**	**Análisis**
Creado por	**Renato Mosquera**	**En revisión**	**Renato Mosquera**
Fecha Creación	**28/01/2015**	**Fecha de Actualización**	**11/02/2015**
Identificador	**RNF003**		
Tipo de Requerimiento	**Alto**	**Tipo de Requerimiento**	**No Funcional**
Datos de Entrada	Cédula del usuario (Directora o Secretaria)		
Descripción	El usuario (Directora o Secretaria) debe ingresar al sistema y podrá enviar la factura electrónica ya validad a los correos electrónicos de los representantes financieros de los alumnos.		
Datos de salida	Factura en archivo de texto.		
Resultados Esperados	Los resultados esperados con este requerimiento es dar la facilidad que los usuarios envíen las facturas a los representantes financieros de los alumnos.		
Origen	Dirección		
Dirigido a	Usuarios (Directora o Secretaria)		
Prioridad	5		
Requerimientos Asociados	RF005		
ESPECIFICACIÓN			
Precondiciones	1. Para ejecutar el requerimiento primero el usuario debe estar registrado en el sistema 2. El usuario debe ser Directora o Secretaria. 3. Una vez ingresado al sistema el usuario podrá enviar la factura electrónica RF005.		
Poscondiciones	Si el usuario (Directora o Secretaria) no recuerda de su clave tendrá que pedir al administrador que le asigne otra clave RNF002. 2.Si el usuario es (Directora o Secretaria) podrá tener acceso a esta opción		
Criterios de Aceptación	Permite que un usuario (Directora o Secretaria) él envió de las facturas a los correos electrónicos de los representantes financieros.		

2.02 Mapeo de Involucrados

Al realizar el levantamiento de los requerimientos se pudo constatar que existen varios usuarios y personajes que intervienen de forma directa o indirecta en este proceso.

- La directora del Centro de Desarrollo Infantil “Descubrir” que se encarga de la parte académica y contable de la institución ya que ella a sus ves es la propietaria del centro educativo.
- La Secretaria que es la encargada del manejo de la parte del registro de la información de las facturas y de realizar los trámites con otras instituciones o personas.
- El Servicio de Rentas Internas (SRI) que es el ente que regula la parte tributaria en el país y realiza la validación de la información que contienen los comprobantes electrónicos en este caso la factura electrónica.
- Padres de familia que son las personas que realizan los pagos de matrículas y pensiones por el servicio educativo que reciben sus hijos en el CDI.
- Los alumnos que son la razón de ser de la institución ya que ellos son los que adquieren el conocimiento y se divierten en las instalaciones del centro educativo.

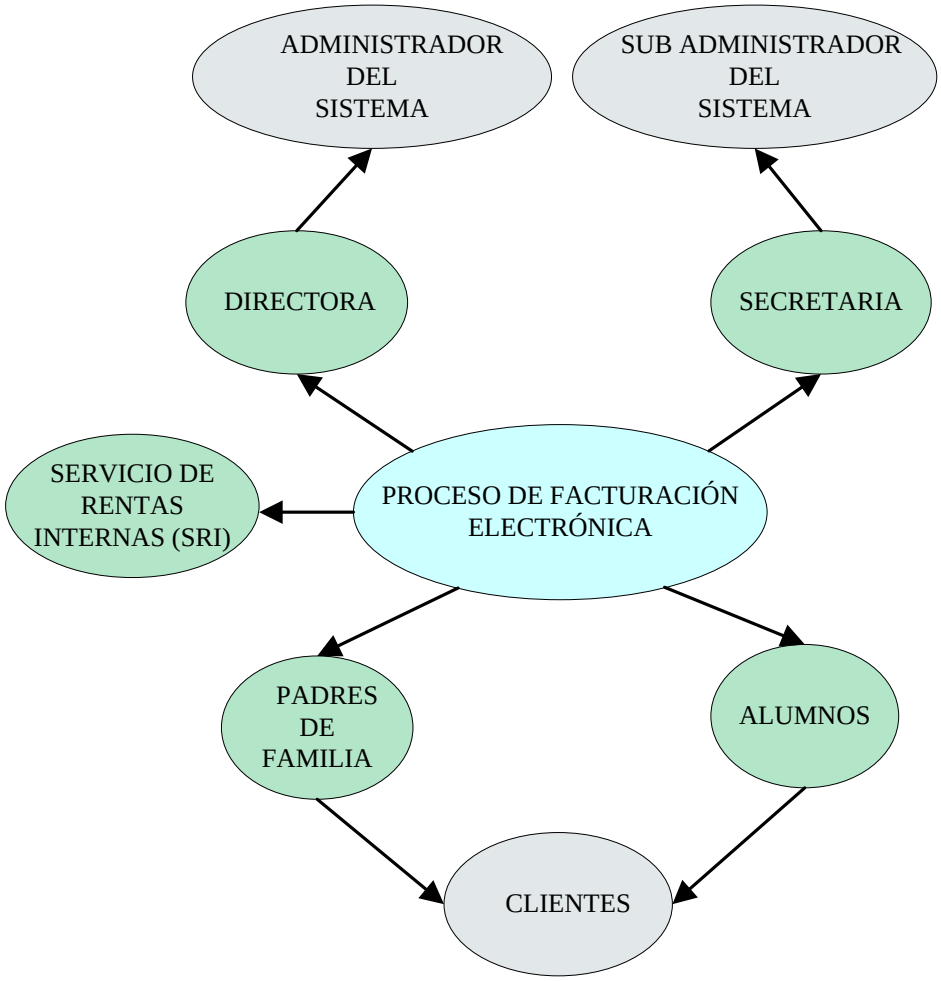


Figura 1 **Detalle de los involucrados con sus relaciones**

2.03 Matriz de Involucrados

En esta matriz se detalla cada uno de los involucrados en el proceso de facturación electrónica de la institución. (Ver anexo A.03)

Capítulo III: Problemas y Objetivos

3.01 Árbol de Problemas

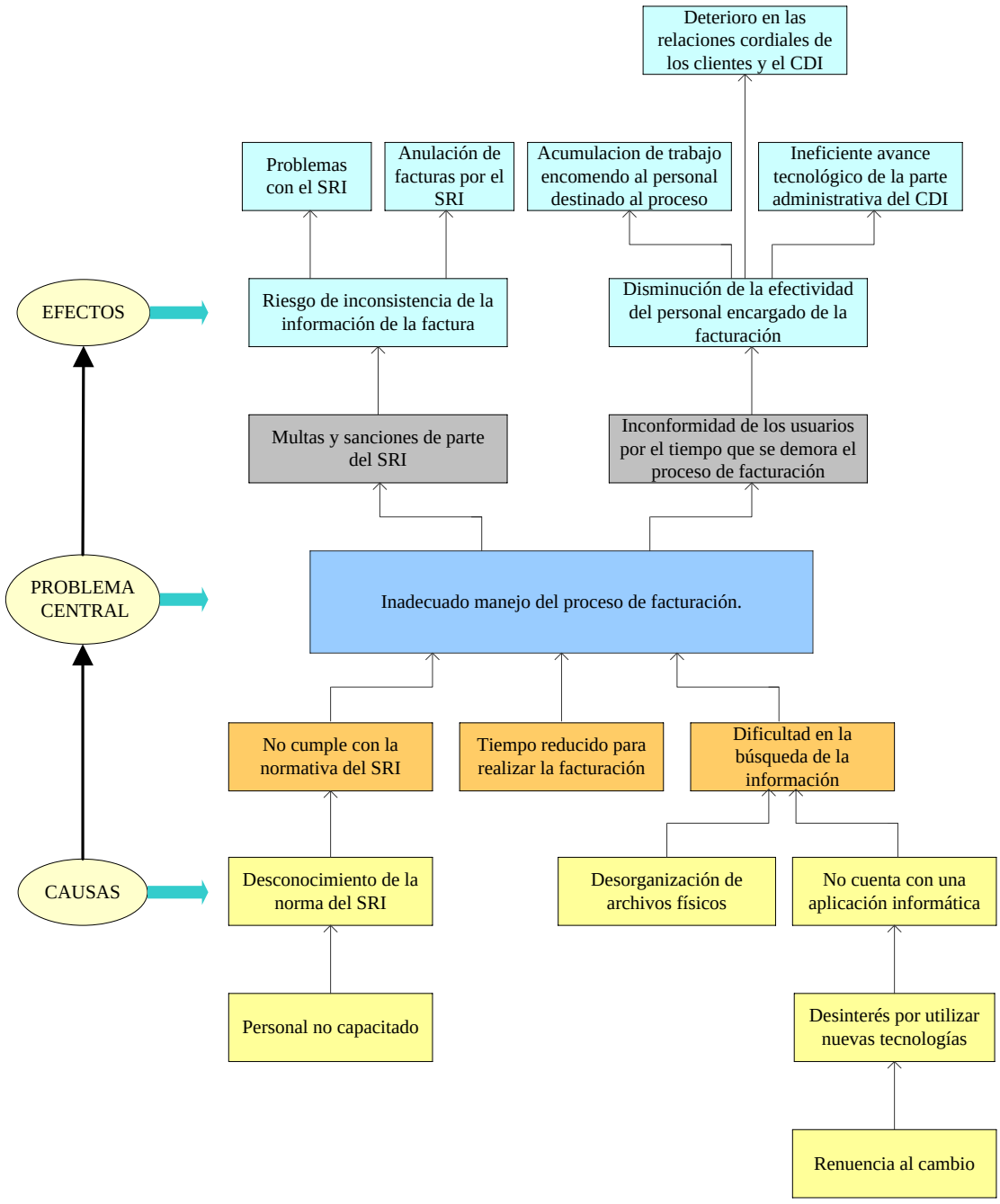


Figura 2 **Organización de Causales, Problema Central y Efectos**

3.02 Árbol de Objetivos

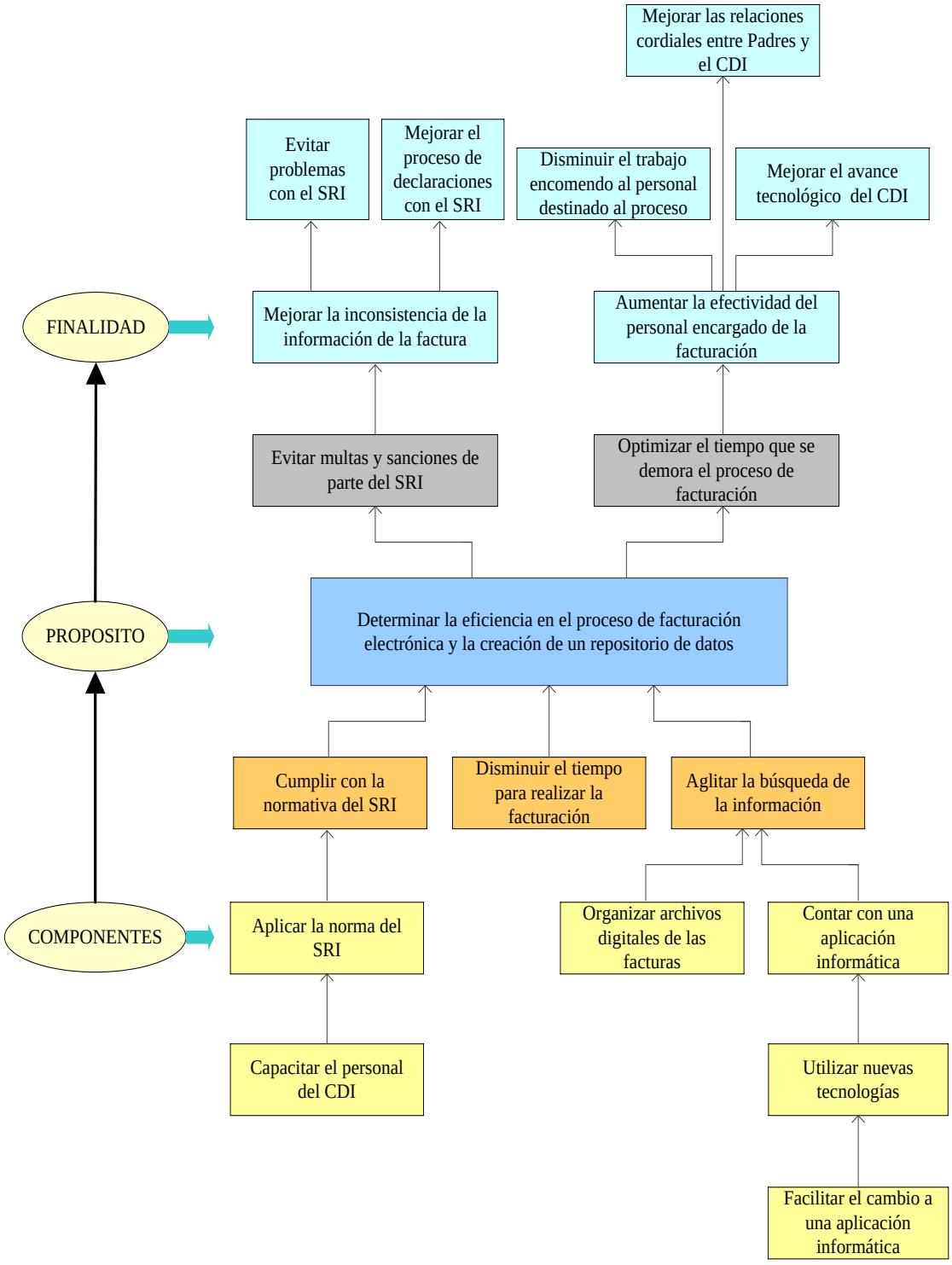


Figura 3 **Detalle de objetivos para llegar a una conclusión y finalidad**

3.03 Diagrama de Casos de Uso

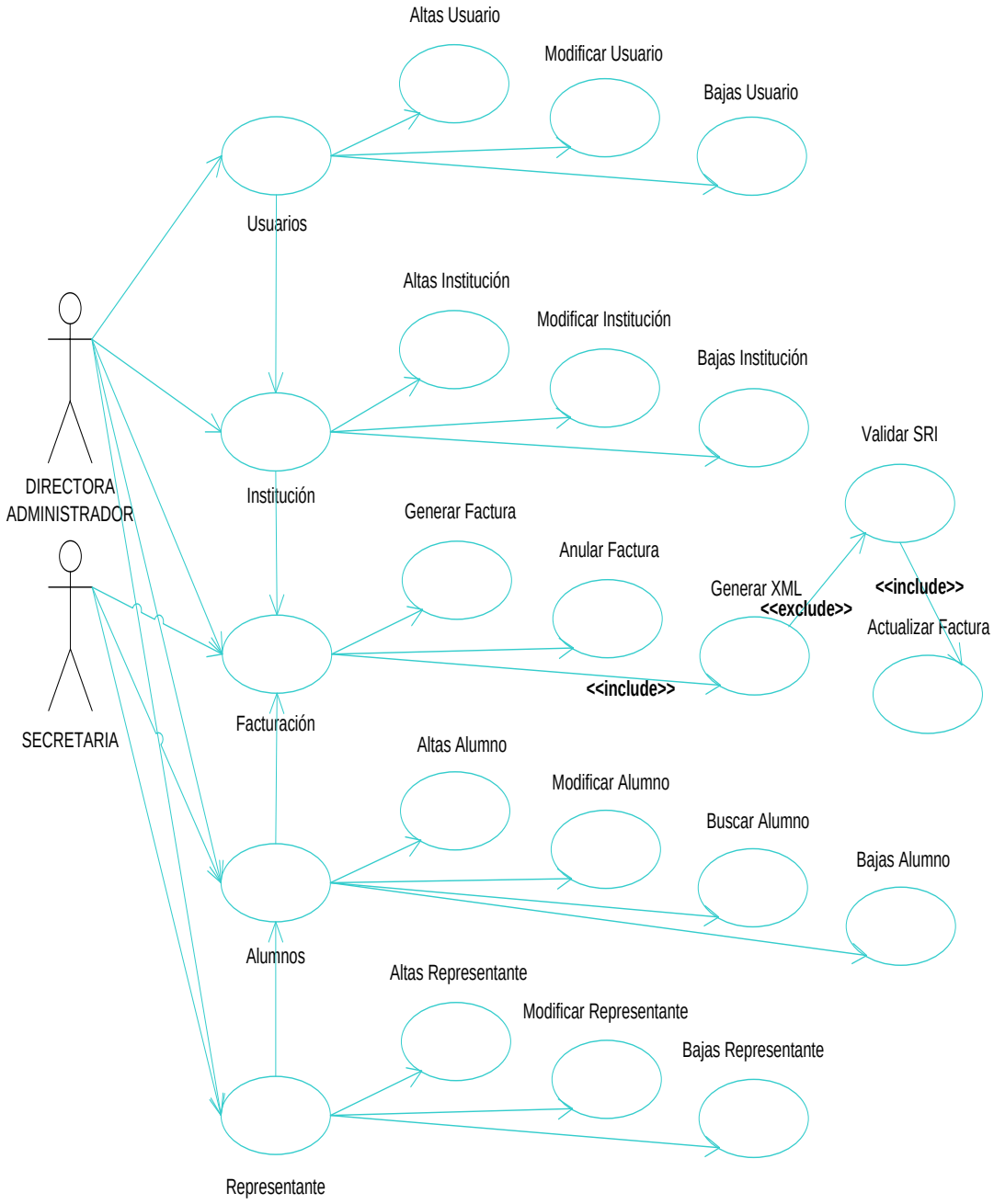


Figura 4 **Caso de uso de Contexto (General) CU001**

3.04 Especificación de Casos de Uso

Tabla 13

Especificación Caso de Uso de Contexto (General) CU001

Caso de Uso 1	Gestión de Usuarios
Identificador	CU001
CURSO TIPICO DE EVENTOS	
Usuario	Sistema
Ingresa al sistema por medio de un login el administrador, elige la opción Insertar Usuario si aún no tiene un usuario y un password. Ingreso de usuario y password, se administrara desde un menú: Usuario, Institución, Representante, Alumno y Facturación Electrónica	Al ingresar en el menú a Usuario se puede realizar alta al usuario en la base de datos siempre y cuando los datos obligatorios se encuentren llenos, también se modificará y se dará de baja al usuario. A esta opción solo tendrá acceso el Administrador del sistema o la directora. Al ingresar a Institución se puede realizar alta a la institución en la base de datos siempre y cuando los datos obligatorios se encuentren llenos, también se modificará y se dará de baja a la institución. A esta opción solo tendrá acceso el Administrador del sistema o la directora. Al ingresar en el menú a Representante se puede realizar alta al representante en la base de datos siempre y cuando los datos obligatorios se encuentren llenos, también se modificará y se dará de baja al representante. A esta opción tendrá acceso el Administrador del sistema o la directora y la secretaria. Al ingresar en el menú a Representante se puede realizar alta al representante en la base de datos siempre y cuando los datos obligatorios se encuentren llenos, también se modificará y se dará de baja al representante. A esta opción tendrá acceso el Administrador del sistema o la directora y la secretaria. Al ingresar en el menú a Representante se puede realizar alta al representante en la base de datos siempre y cuando los datos obligatorios se encuentren llenos, también se modificará y se dará de baja al representante. A esta opción tendrá acceso el Administrador del sistema o la directora y la secretaria. Al ingresar en el menú a Facturación Electrónica se puede generar la factura electrónica, también se puede anular la factura electrónica y generar el XML siempre y cuando los datos necesarios de Institución, Usuario, Representante, Alumno estén correctos, para luego enviar para la respectiva validación del SRI. Al recibir la factura electrónica ya validada se procede a actualizar la factura electrónica. A esta opción tendrá acceso el Administrador del sistema o la directora y la secretaria.
CURSOS ALTERNATIVOS	
Curso Alternativo de eventos 1	

3.05 Casos de Uso de Realización

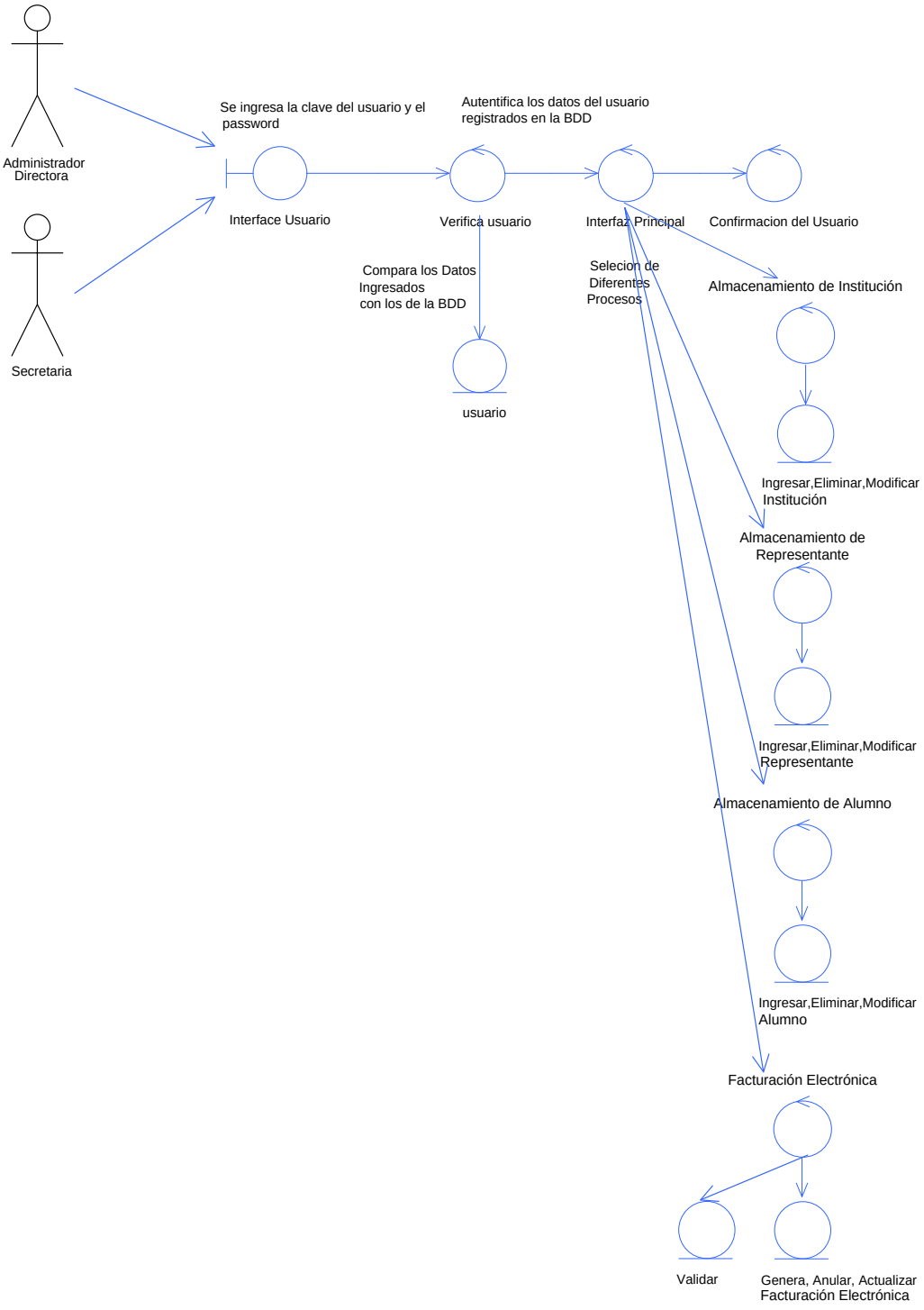


Figura 5 **Caso de uso de realización General CUR001**

3.06 Especificación de Casos de uso de realización

Tabla 14

Especificación Caso de Uso de Realización CUR001

Nombre	**Administración del Proceso**
Identificador	**CUR001**
Responsabilidades	Realizar la creación de usuarios, primero valida, si no existe procede a crear. Realizar manejo Institución ingresa los datos y precede a validar. Realizar el manejo de Representante ingresa los datos y precede a validar. Realizar manejo de Alumno ingresa los datos y precede a validar. Todos estos menús tendrán la posibilidad de ejecutar los mantenimientos de: Ingresar, Modificar, Buscar, Eliminar Realizar Manejo de Facturación Electrónica se carga los datos de institución, usuario, alumno, representante y se genera un XML para la validación por el SRI Después que regresa el XML validado se actualiza la factura. Se podrá realizar en este menú: Generar, anular, actualizar factura electrónica.
Tipo	Sistema
Referencias Casos de uso	CU001
Referencias requisitos	No tiene
PRECONDICIONES	
De Instancia	
Se necesita una interface para que el administrador ingrese el nuevo usuario. Se necesita una entidad para guardar la información de institución, representante, alumno, factura electrónica.	
De Relación	
No tiene	
POSCONDICIONES	
SALIDAS PANTALLA	
Registro Guardado correctamente de institución, representante, alumno. XML generado correctamente.	

3.07 Diagramas de secuencia del sistema

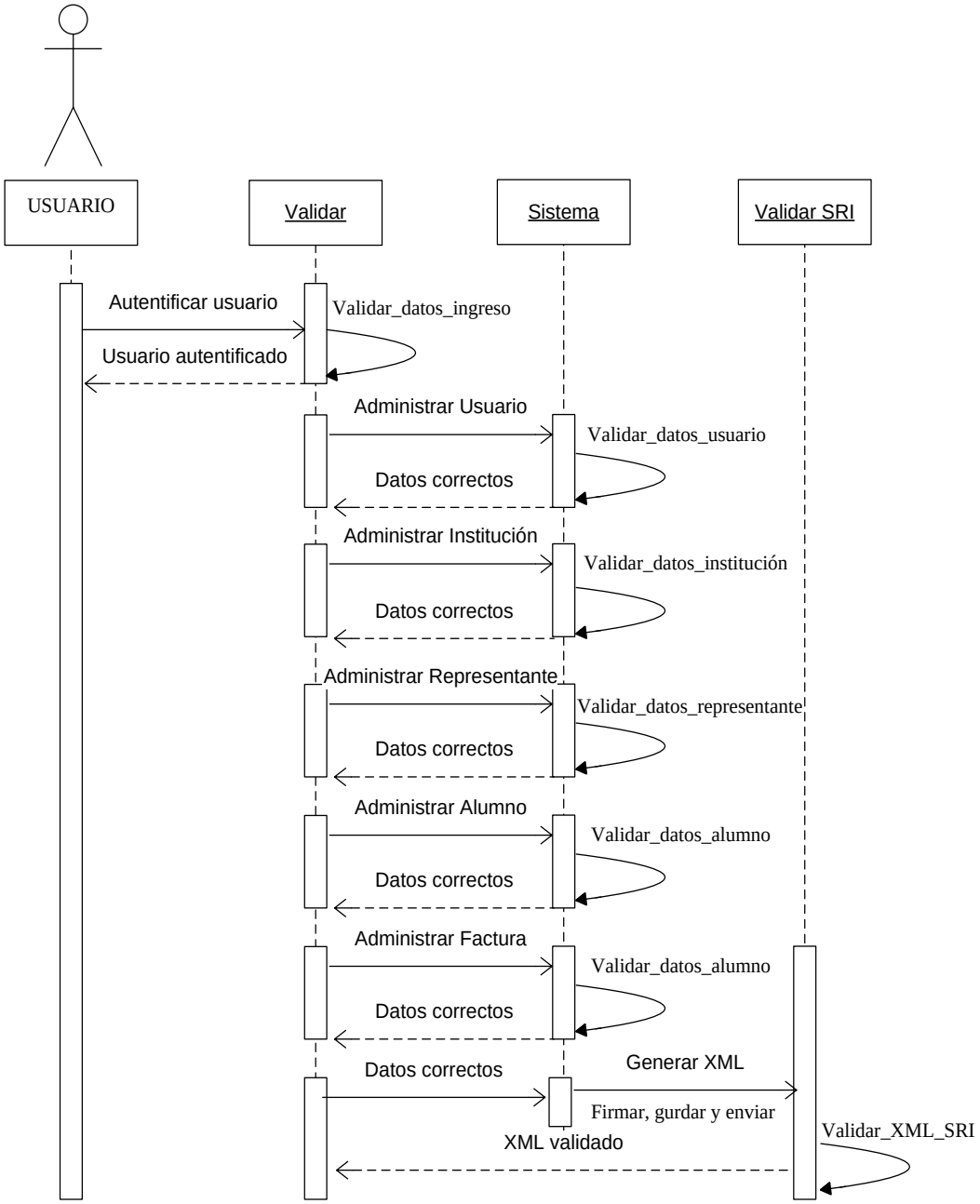


Figura 6 **Diagrama de Secuencia General**

Capítulo IV: Análisis de Alternativas

4.01 Matriz de Análisis de Alternativas

Tabla 15

Análisis estadístico de alternativas

Objetivos	Impacto sobre el propósito	Factibilidad Técnica	Factibilidad Financiera	Factibilidad Social	Factibilidad Política	Total	Categoría
Cumplir con la normativa del SRI	5	5	3	3	2	18	Alta
Disminuir el tiempo para realizar la facturación	5	5	3	3	2	18	Alta
Agilitar la búsqueda de la información	5	4	4	3	2	18	Alta
Organizar archivos digitales de las facturas	5	5	2	2	2	16	Media Alta
Contar con una aplicación informática	5	5	3	3	2	18	Alta
TOTAL:	25	24	15	14	10	88	

Tabla 16

Detalle de valores del análisis de alternativas

CATEGORIAS	VALORES	PORCENTAJES
Alta	4	80%
Media Alta	1	20%
Media Baja	0	0%
Baja	0	0%
TOTAL:	5	100%

4.02 Matriz de Impacto de Objetivos

Tabla 17

Análisis de los objetivos

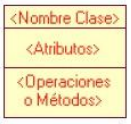
Objetivos	*Factibilidad de Lograrse* (A-M-B) (4-2-1)	*Impacto en Género* (A-M-B) (4-2-1)	*Impacto Ambiental* (A-M-B) (4-2-1)	*Relevancia* (A-M-B) (4-2-1)	*Sostenibilidad* (A-M-B) (4-2-1)	*Total* Puntos
Cumplir con la normativa del SRI	Se beneficiara ya que cumple la norma	Es beneficioso para cualquier genero	Fomenta el reciclaje al guardar las facturas en papel	Responde a las expectativas del CDI	Ejecuta la norma en el campo educativo	
Disminuir el tiempo para realizar la facturación	Es de gran conveniencia para la institución	Disminuye el tiempo de respuesta	Mejora el entorno ambiental y social	Es una prioridad manifiesta del CDI	Fortalece la organización dentro del CDI	**0 a 5 BAJA**
Agilitar la búsqueda de la información	Cuenta con la tecnología necesaria realizar este cambio	Podrá utilizar personal de cualquier género		Beneficia a todos los involucrados en el proceso	Fortalece la forma de realizar el proceso	**6 a 10 MEDIA BAJA**
Organizar archivos digitales de las facturas	Se minimizará tiempo empleado en el proceso	Podrá buscar la información el personal de cualquier género	Protege el uso de los recursos naturales	Los beneficios son los esperados por el CDI	Tendrá gran capacidad de almacenamiento	**11 a 15 MEDIA ALTA**
Contar con una aplicación informática	El desarrollo del sistema que su beneficio será mayor que su costo	Se podrá utilizar por personal de cualquier género	Mejora la educación ambiental de los beneficiarios	Los usuarios quedarán con un alto grado de conformidad	Se cumplirá con los requerimientos del CDI	**16 a 20 ALTA**
Puntos	20	13	15	20	11	79

4.03 Estándares para el Diseño de Clases

4.03.01 Clases

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

En UML, una clase es representada por un rectángulo que posee tres divisiones:



Superior: Contiene el nombre de la Clase

Intermedio: Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).

Inferior: Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

4.03.02 Atributos y Métodos:

Atributos.- Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:

public (+,): Indica que el atributo será visible tanto dentro como fuera de la clase, es decir, es accsesible desde todos lados.

private (-,): Indica que el atributo sólo será accesible desde dentro de la clase (sólo sus métodos lo pueden accesar).

protected (#,): Indica que el atributo no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de las subclases que se deriven.

4.03.03 Métodos

Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, éstos pueden tener las características:

public (+,): Indica que el método será visible tanto dentro como fuera de la clase, es decir, es accsesible desde todos lados.

private (-,): Indica que el método sólo será accesible desde dentro de la clase (sólo otros métodos de la clase lo pueden accesar).

protected (#,): Indica que el método no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de métodos de las subclases que se deriven.

4.04 Diagrama de Clases

Diagrama orientado a objetos que muestra las clases.

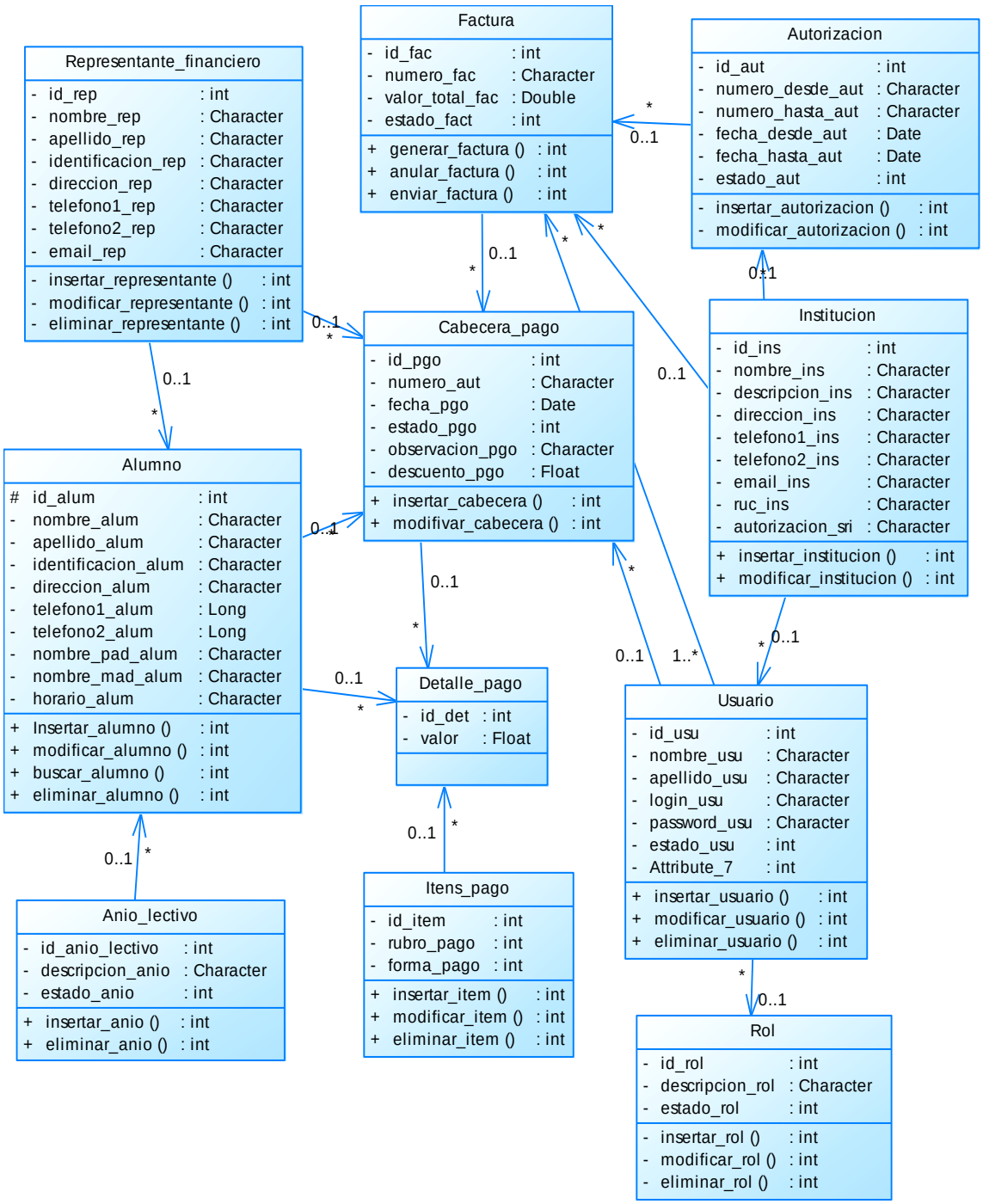


Figura 7 **Donde se especifica los atributos y acciones del sistema**

4.05 Modelo Lógico y Físico

4.05.01 Modelo Lógico

Es un diagrama lógico de la base de datos.

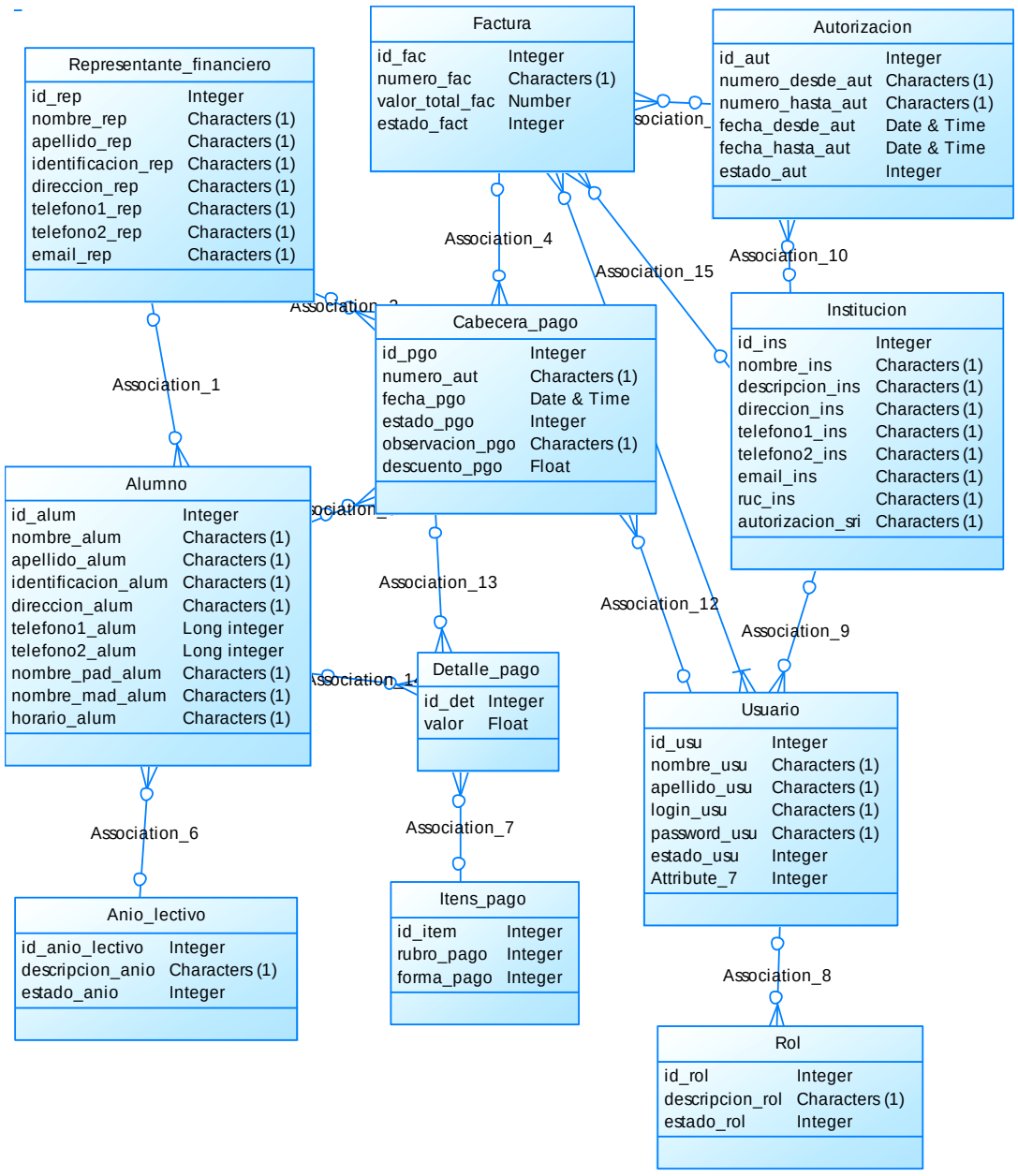


Figura 8 **Donde se agrega la persistencia de cada clase**

4.05.01 Modelo Físico

Es el esquema físico de cómo está conformada la base de datos.

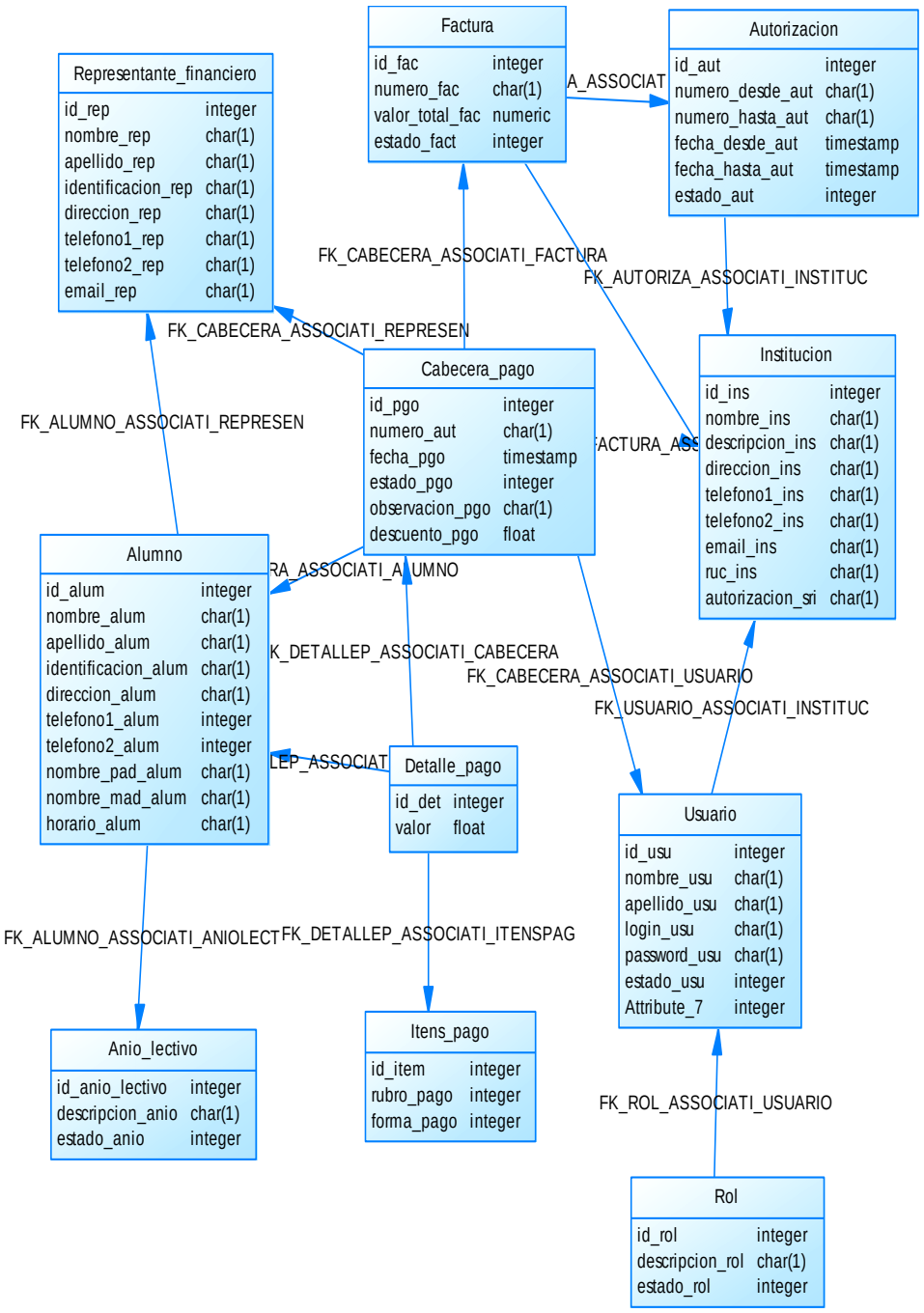


Figura 9 **Donde se genera las tablas en base a los diagramas anteriores**

4.06 Diagrama de Componentes

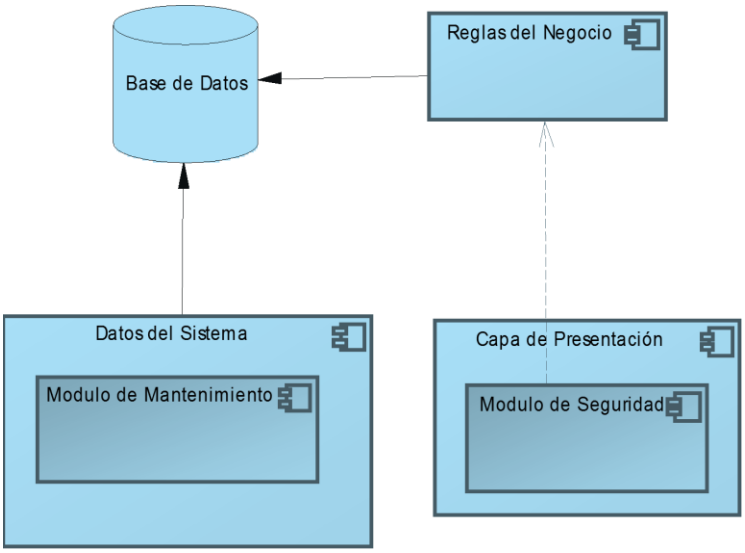


Figura 10 **Donde se especifica los módulos y las capas del sistema**

4.07 Diagrama de Estrategias

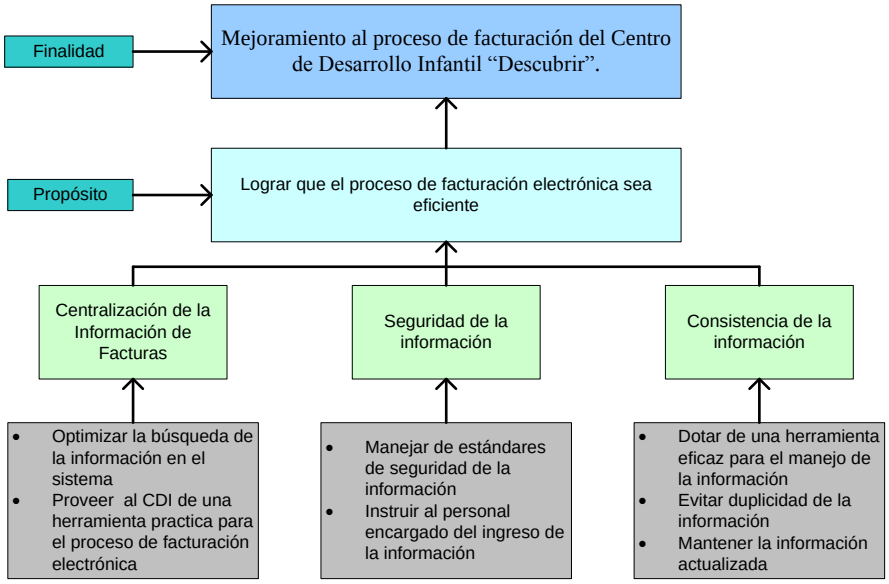


Figura 11 Donde se especifica las estrategias para llegar a una finalidad

4.08 Matriz de Marco Lógico

Tabla 18

Resumen del proyecto que destaca lo que se desea lograr

Resumen narrativo de objetos	Indicadores	Medios de verificación	Supuestos
Fin: Mejoramiento al proceso de facturación del Centro de Desarrollo Infantil "Descubrir".	Controlar la inconsistencia de la información de la factura mediante validaciones de datos	Al realizarse la validación de los datos por el SRI se mostrara que se cumple con todos los requerimientos de la institución y parámetros del SRI.	Capacitación de los encargados del proceso para que utilicen herramientas tecnológicas.
Propósito: Determinar la eficiencia en el proceso de facturación electrónica y la creación de un repositorio de datos	Evitar la inconformidad de los usuarios por el tiempo que se demora el proceso de facturación gracias a una atención rápida.	Al tener reportes que muestren los datos requeridos por la institución se cumple con los propósitos de la aplicación.	Realizar mantenimiento periódicos de la aplicación para un correcto desempeño.
Componentes: 1. Cumplir con la normativa del SRI 2. Disminuir el tiempo para realizar la facturación 3. Agilitar la búsqueda de la información	Centralización de la Información de Facturas, Seguridad de la información, Consistencia de la información todo esto se puede verificar en el repositorio datos	Realizar medios para que la búsqueda de la información de los alumnos sea rápida y consistente.	Tener en cuenta las restricciones que debe tener la aplicación para el manejo de la información.
Actividad: 1. Optimizar la búsqueda de la información en el sistema 2. Proveer al CDI de una herramienta práctica para el proceso de facturación electrónica 3. Manejar de estándares de seguridad de la información 4. Instruir al personal encargado del ingreso de la información 5. Dotar de una herramienta eficaz para el manejo de la información 6. Evitar duplicidad de la información	Los gatos que se generaron para la elaboración de las actividades principalmente son en movilidad 50$, en hospedaje para el sitio 30$, Tres resmas de papel 13,50$, Obtención de la firma electrónica 100$.	Documentar todo el proceso desde la entrevista, el levantamiento de la información, la elaboración de la base de datos y el desarrollo de la aplicación como respaldo de la realización del sistema.	El tiempo empleado para la realización del proyecto es reducido ya que se necesita barias visitas al CDI y también ir a la par realizando la aplicación y el documento escrito.

4.09 Vistas Arquitectónicas

4.09.01 Vista Lógica

Describe el modelo de objetos del diseño del sistema.

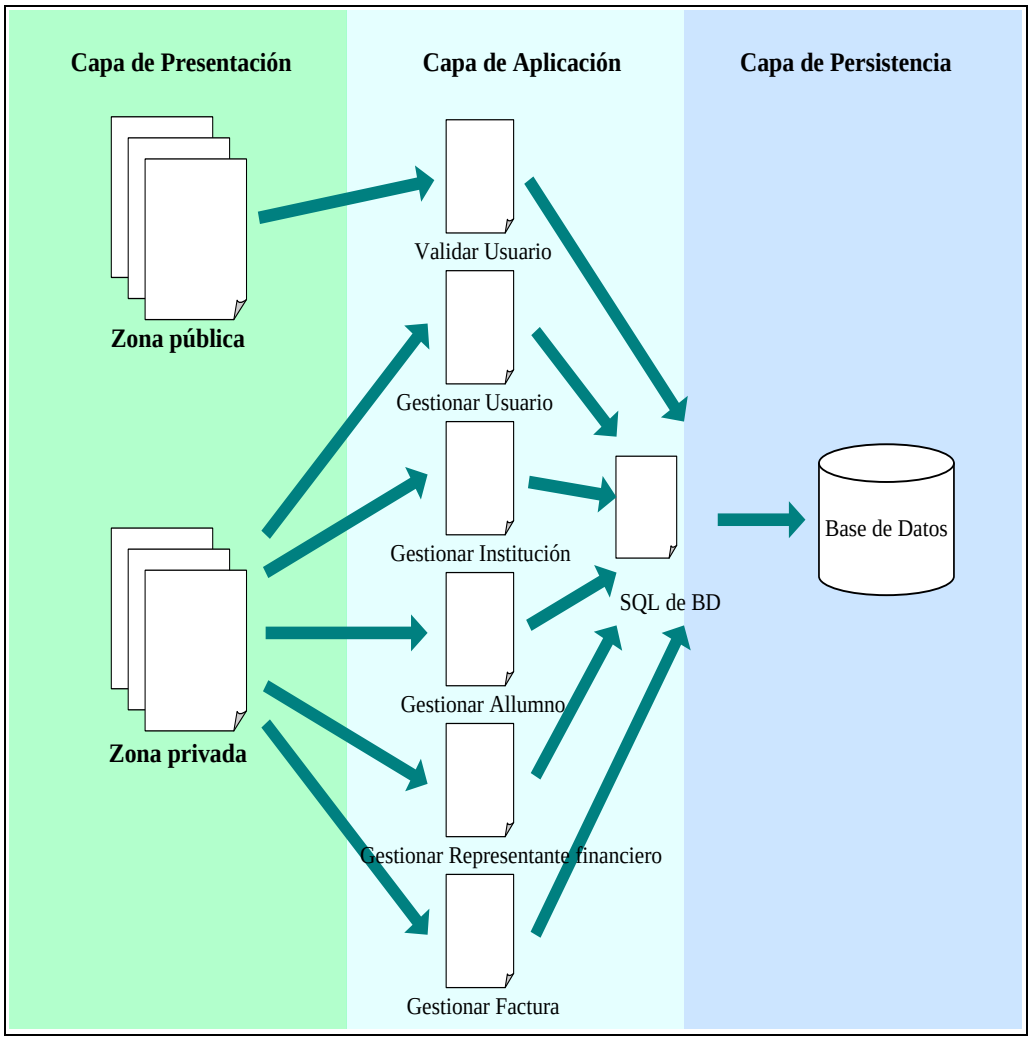


Figura 12 **Descripción de la lógica del sistema**

4.09.02 Vista Física

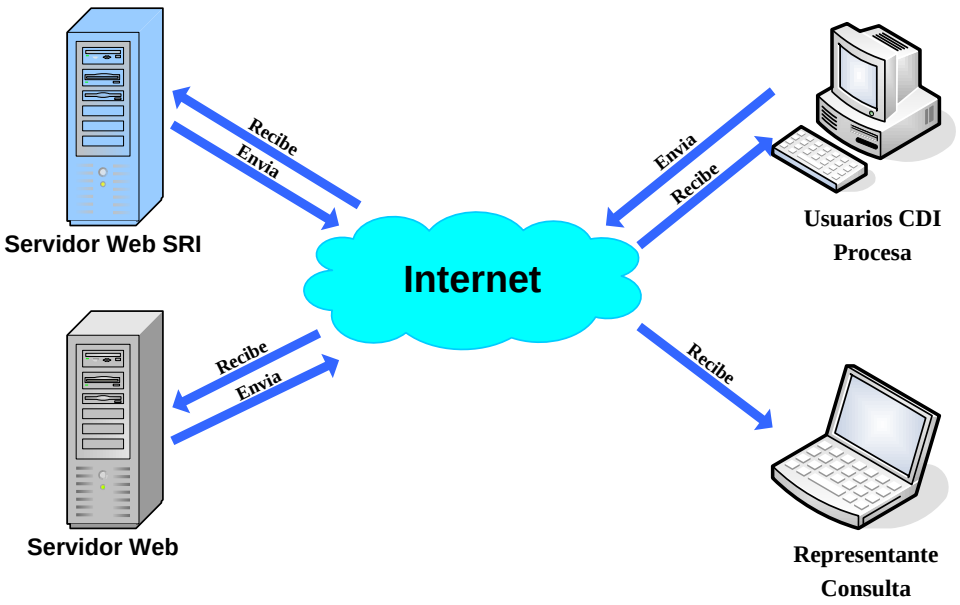


Figura 13 **Descripción física del sistema**

4.09.03 Vista de Desarrollo

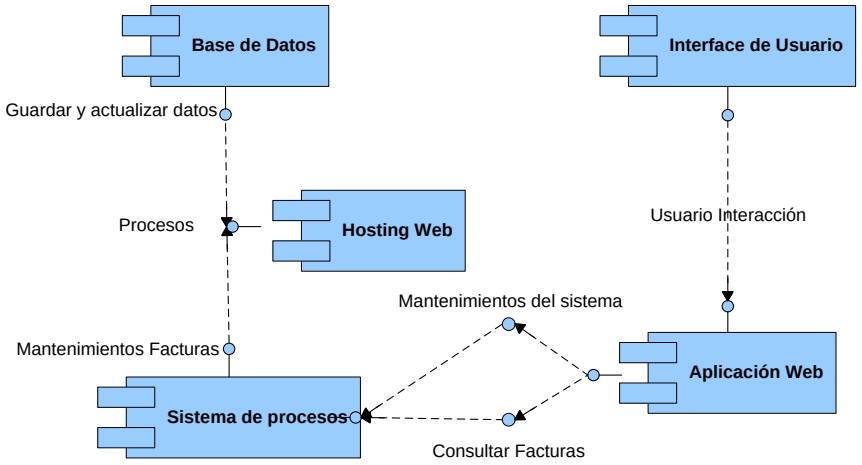


Figura 14 **Descripción detallada del sistema mediante componentes**

4.09.04 Vista de Procesos

Es donde se describe todos los proceso de ingreso de datos de la aplicación.

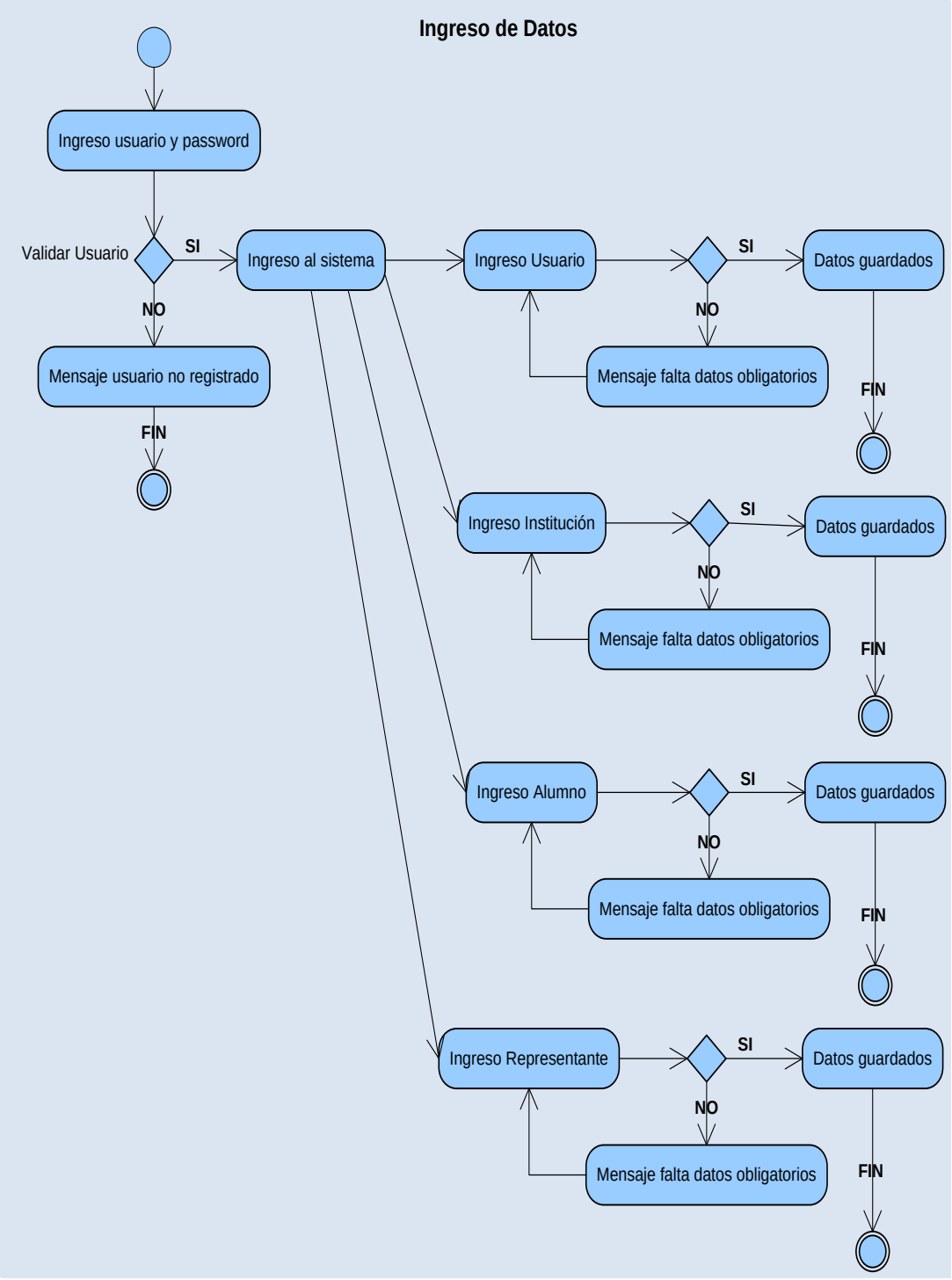


Figura 15 **Vista del Proceso Ingreso de Datos**

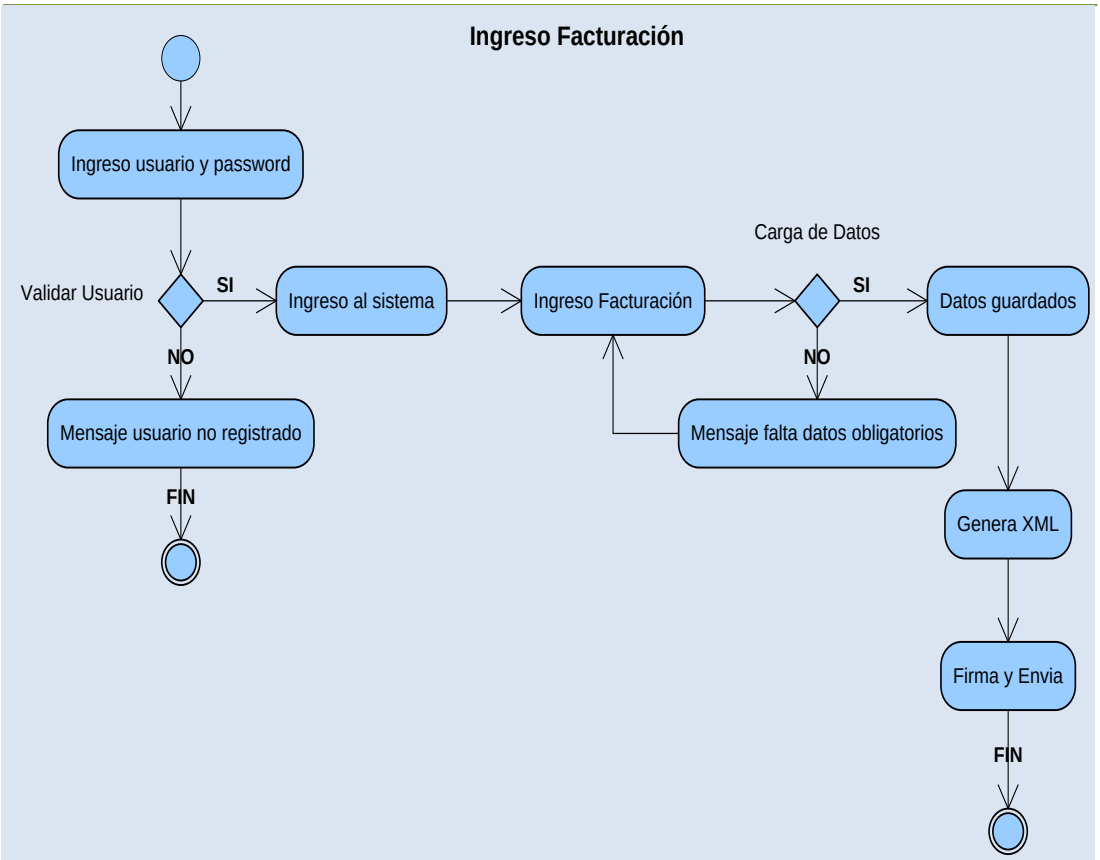


Figura 16 **Vista del Proceso Facturación**

Capítulo V: Análisis de Alternativas

5.01 Especificación de estándares de programación

Estos estándares son de gran utilidad ya que permiten saber y entender cómo se ingresó el código fuente del sistema, las variables, ficheros, nombres de tablas y atributos en la base de datos, etc.

5.01.01 Declaraciones de Variables

La declaración de las variables, se ajusten al motivo para la que se requieran. El mnemotécnico definido se establece tomando en consideración principalmente lo siguiente:

- La longitud debe ser lo más recomendable posible.
- El tipo de dato al que pertenece la variable.
- El tipo se determina en función de los tipos de las expresiones que se indican como argumentos de la operación.
- Una variable se declara usando la misma sintaxis de C

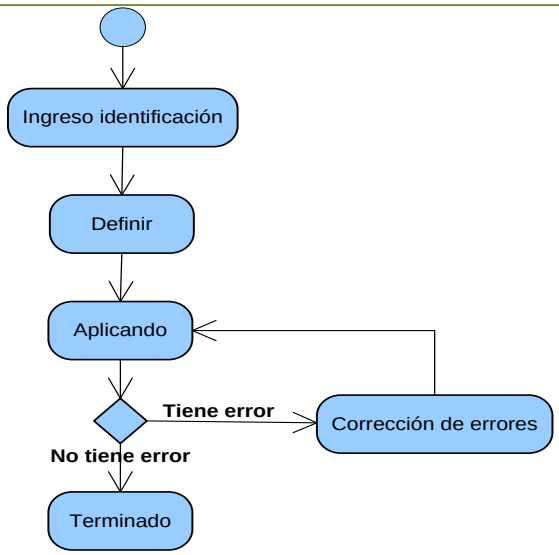


Figura 17 **Diagrama de secuencia del proceso de pruebas de aceptación**

Por lo tanto la estructura de la variable es como sigue:

Siendo el nombre que identifica a la variable: $a

Tabla 19

Detalle de la descripción de un variable

Estructura	Descripción de la Variable
LONGITUD. MAX.	← 1 →← 16 →
FORMATO	Todo con minúsculas
EJEMPLO	$meses, $a = 1; $cadena="Hola amigo";

5.01.02 Descripción de la variable

Nombre que se le asignará a la variable para que se le identifique y deberá estar asociada al motivo para la cual se le declara.

Ejemplo: $a = 1; $cadena="Hola amigo"; $sum = “suma”;

5.01.03 Variables de Tipo Arreglo

En el caso de las definiciones de arreglos de elementos se declarara la variable con el prefijo de “lista”, el cual nos dará entender que se trata de una variable del tipo arreglo la cual contendrá de cero a más datos, según el tamaño declarado.

Ejemplos: lista_meses

$lista_meses=

array("ENE","FEB","MAR","ABR","MAY","JUN","JUL","AGO","SEP","OCT","NOV","DIC");

5.01.04 Definición de Controles

Para poder determinar el nombre de un control adentro de cualquier aplicación de tipo visual, se procede a identificar el tipo al cual pertenece y la función que cumple adentro de la aplicación.

Tipo de datos

Tabla 20

Detalle del tipo de datos de una variable

Tipo de variable	Mnemónico	Descripción
Byte	by	Entero de 8 bits sin signo.
Integer	in	Entero de 32 bits con signo.
Char	ch	Un carácter UNICODE de 16 bits
String	st	Cadena de caracteres
Date	dt	Formato de fecha/hora
Boolean	bl	Valor lógico: verdadero y falso
Float	fl	Coma flotantes, 11-12 dígitos significativos.
Double	db	Coma flotante, 64 bits (15-16 dígitos significativos)
Object	ob	Objeto genérico

Prefijo para el Control

El prefijo del control será determinado mediante tres caracteres que estarán conformados por las consonantes más representativas del control, es así, por ejemplo; el control Button, estará asociado al prefijo btn.

Nombre descriptivo del Control

Formado por la descripción de la función que lleva a cabo el control, esta debe ser descrita en forma específica y clara.

Tabla 21

Detalle del nombre descriptivo para controles

Tipo de control	Prefijo	Ejemplo
Label	lbl	lblNombre
TextBox	txt	txtApellido
Button	btn	btnLogin
RadioButton	rdo	rdoSeleccion
CheckBox	chk	chkRuta1
DropDownList	cmb	cmbDocumentos

Declaración de variables, atributos y objetos

Se debe declarar una variable por línea.

Tabla 22

Detalle de como declarar variables, atributos y objetos

Título	Descripción
Sintaxis	[Tipo Variable] [Nombre de la Variable]
Descripción	Las variables y atributos tendrán una longitud máxima de 50 caracteres. El nombre de la variable puede incluir más de un sustantivo los cuales se escribirán juntos. Si se tuvieran variables que puedan tomar nombres iguales, se le agregará un número asociado (si está adentro de un mismo método será correlativo).
Observaciones	La declaración de variables o atributos no se deberá utiliza caracteres como: • Letra Ñ o ñ & caracteres especiales ¡ ^, #, $, %, &, /, (,), ¿, ‘, +, -, *, {, }, [,]. • Caracteres con tildes: á, é, í, ó, ú.
Ejemplo	Public String nombre Indica una variable o atributo que guardará un nombre.

Declaración de Clases

Tabla 23

Detalle de como declarar clases

Título	Descripción
Sintaxis	[Tipo] Class [Nombre de Clase]
Descripción	El nombre de las clases tendrá una longitud máxima de 50 caracteres y las primeras letras de todas las palabras estarán en mayúsculas. Tipo se refiere a si la clase será: Private, Public o Protected.
Observaciones	La declaración de clases no se debe utilizar caracteres como: • Letra Ñ o ñ & caracteres especiales ¡ ^, #, $, %, &, /, (,), ¿, ‘, +, -, *, {, }, [,]. • Caracteres con tildes: á, é, í, ó, ú.
Ejemplo	Private Class Alumno Indica una clase Alumno

Declaración de Métodos

Tabla 24

Detalle de como declarar Métodos

Título	Descripción
Sintaxis	NombreProcedim[(ListaParámetros)]
Descripción	El nombre del método constará hasta de 30 caracteres. La primera letra de la primera palabra del nombre será escrita en minúscula y las siguientes palabras empezarán con letra mayúscula.
Observaciones	La declaración de métodos no se deberá utilizar caracteres como: • Letra Ñ o ñ & caracteres especiales ¡ ^, #, $, %, &, /, (,), ¿, ‘, +, -, *, {, }, [,]. • Caracteres con tildes: á, é, í, ó, ú.
Ejemplo	Protected ingresarRubro (String rubro) Indica un método ingresarRubro que recibe una variable por valor de tipo string al ámbito de la clase

Declaración de Funciones

Tabla 25

Detalle de como declarar Funciones

Título	Descripción
Sintaxis	[TipoDato] nombreFuncion[(ListaParámetros)]
Descripción	El nombre del objeto constará hasta de 30 caracteres, no es necesario colocar un nombre que indique la clase a la cual pertenece. La primera letra de la primera palabra del nombre será escrita en mayúsculas El dato de retorno se coloca al final y será obligatorio colocarlo.
Observaciones	En la declaración de objetos no se deberá utilizar caracteres como: • Letra Ñ o ñ & caracteres especiales ¡ ^, #, $, %, &, /, (,), ¿, ‘, +, -, *, {, }, [,]. • Caracteres con tildes: á, é, í, ó, ú.
Ejemplo	Public int sumar(int A, int B) Indica una función que suma dos variables enteras

5.02 Diseño de Interfaces de Usuario

El diseño de la interfaz de usuario siempre debe estar enfocado a la comodidad y dinámica del usuario ya que esta interfaz será visualizada y utilizada siempre en su jornada de trabajo, en la cual no se debe descuidar los componentes como el hardware y el software con sus distintas características y disposición.

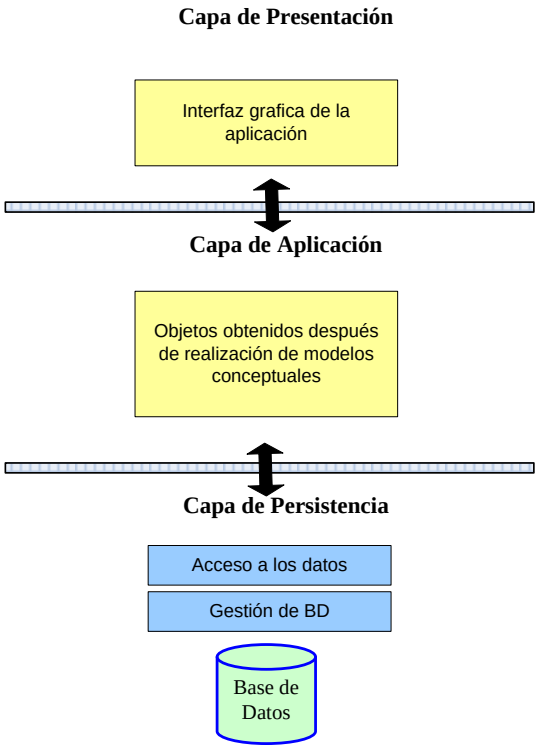


Figura 18 **Capas en donde interactúa la interface del usuario**

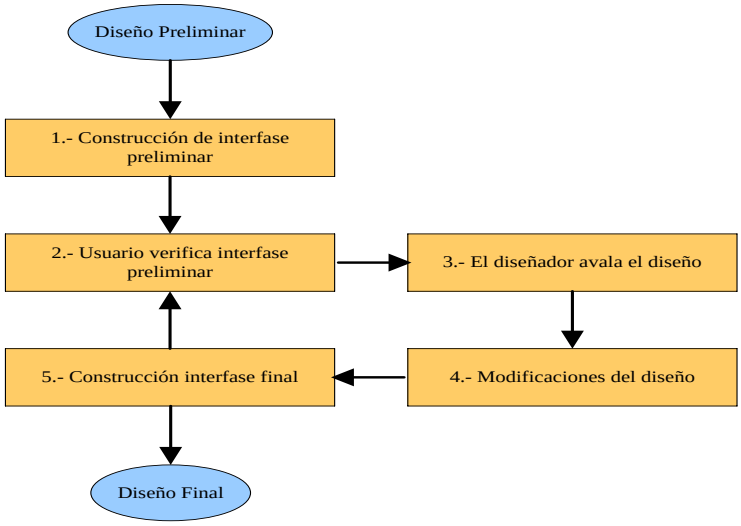


Figura 19 **Diagrama general de los pasos para diseñar una interface**

Figura 20 **Diseño general de interface de usuario**

5.03 Especificación de pruebas de unidad

Una prueba de unidad sirve para probar las funciones en un archivo de programa simple (una clase en terminología de objetos). Las librerías de pruebas de unidad formalizan este trabajo al proporcionar clases para pruebas, esto ayuda a que el módulo se haga independiente. Esto quiere decir que un módulo que tiene una prueba de unidad se puede probar independientemente del resto de la aplicación. Esto se lo realiza verificando o probando cada módulo.

Tabla 26

Prueba de interface de usuario (estándares)

Identificador de la Prueba:	**PRU_UNI001**
Método a Probar	**Interface**
Objetivo de la Prueba	Determinar las fallas en el manejo de la interface y corregir si es necesario, Inspeccionar los distintos estándares utilizados para mejorar la navegación por la aplicación de los usuarios.
Datos de Entrada:	
Datos en generales en los diferentes Formularios empleados en la aplicación	
Resultados Esperados	
Detectar posibles errores que generen un mal funcionamiento del sistema	
Comentarios	

Tabla 27

Pruebas de Reportes, resultados eficientes

Identificador de la Prueba:	**PRU_UNI002**
Método a Probar	**Reportes**
Objetivo de la Prueba	Inspeccionar que todos los resultados esperados por el usuario en el proceso sean los adecuados y correctos.
Datos de Entrada:	
Datos en generales en los diferentes Formularios empleados en la aplicación.	
Resultados Esperados	
Detectar posibles errores que generen un mal funcionamiento al realizar reportes	
Comentarios	
Al realizar pruebas en los reportes se encontró errores únicamente ortográficos	

Tabla 28

Pruebas de compilación de Código

Identificador de la Prueba:	PRU_UNI003
Método a Probar	**Pruebas de Código - Compilación**
Objetivo de la Prueba	Evaluar los resultados obtenidos y analizar los errores del código encontrados para luego corregirlos
Datos de Entrada:	
Compilación de cada proceso paso a paso.	
Resultados Esperados	
Mantener el sistema ejecutándose, corregir para que no exista errores la volver a compilar.	
Comentarios	

Tabla 29

Pruebas de Almacenamiento de datos en la Base

Identificador de la Prueba:	PRU_UNI004
Método a Probar	**Almacenamiento de datos en la base**
Objetivo de la Prueba	Determinar si los datos ingresados en los mantenimientos de todos los formularios están realizándose de la forma correcta.
Datos de Entrada:	
Datos requeridos en los formularios de Mantenimiento de la aplicación	
Resultados Esperados	
Obtener datos que mantengan consistencia y coherencia.	
Comentarios	
Con la prueba se verifica encriptación de contraseñas y que los datos y rubros sean reales al presentar en los diferentes formularios y reportes	

5.04 Especificación de pruebas de aceptación

Tabla 30

Detalle de pruebas de aceptación en la creación de usuarios

Identificador de la Prueba:	PRU_ACE001
Caso de Uso	**Usuarios CU001**
Tipo de Usuario	**Administrador**
Objetivo de la Prueba	Comprobar el funcionamiento del proceso Administrar usuarios.
Secuencia de Eventos	
Login de usuario, Ingresar Usuarios, Modificar y Eliminar.	
Resultados Esperados	
No exista inconsistencias de los datos guardados, validaciones y seguridad en la información.	
Comentarios	
Se realizó las pruebas ingresando un nuevo usuario en el formulario y dando mantenimiento a la tabla.	
Estado Aceptado/No aceptado	
Aceptado	

Tabla 31

Detalle de pruebas de aceptación en el proceso de Institución

Identificador de la Prueba:	PRU_ACE002
Caso de Uso	**Institución CU002**
Tipo de Usuario	**Administrador/Directora**
Objetivo de la Prueba	Comprobar el funcionamiento del proceso de administración de institución.
Secuencia de Eventos	
Login de usuario, ingresar datos generales de la Institución.	
Resultados Esperados	
Que no tenga inconsistencias con respecto a los datos guardados, validaciones y seguridad en la información de la institución.	
Comentarios	
Las pruebas en la aplicación ingresando datos, eliminando, modificando.	
Estado Aceptado/No aceptado	
Aceptado	

Tabla 32

Detalle de pruebas de aceptación en el proceso de Alumno

Identificador de la Prueba:	**PRU_ACE003**
Caso de Uso	**Alumno CU004**
Tipo de Usuario	**Administrador/Directora, Secretaria**
Objetivo de la Prueba	Comprobar el funcionamiento del proceso de administración de Alumno.
Secuencia de Eventos	
Login de usuario, ingresar datos generales del Alumno	
Resultados Esperados	
Que no tenga inconsistencias con respecto a los datos guardados, validaciones y seguridad en la información del Alumno	
Comentarios	
Las pruebas en la aplicación ingresando datos, eliminando, modificando y buscando.	
Estado Aceptado/No aceptado	
Aceptado	

Tabla 33

Detalle de pruebas de aceptación en el proceso de Representante

Identificador de la Prueba:	**PRU_ACE004**
Caso de Uso	**Representante CU005**
Tipo de Usuario	**Administrador/Directora, Secretaria**
Objetivo de la Prueba	Comprobar el funcionamiento del proceso de administración de Representante.
Secuencia de Eventos	
Login de usuario, ingresar datos generales del Representante	
Resultados Esperados	
Que no tenga inconsistencias con respecto a los datos guardados, validaciones y seguridad en la información del Representante	
Comentarios	
Las pruebas en la aplicación ingresando datos, eliminando, modificando.	
Estado Aceptado/No aceptado	
Aceptado	

Tabla 34

Detalle de pruebas de aceptación en el proceso de Facturación Electrónica

Identificador de la Prueba:	**PRU_ACE05**
Caso de Uso	**Facturación Electrónica CU003**
Tipo de Usuario	**Usuario**
Objetivo de la Prueba	Probar el funcionamiento en la generación de factura y del XML
Secuencia de Eventos	
Login de usuario, Factura Electrónica.	
Resultados Esperados	
Que no tenga inconsistencias los datos y los rubros sean reales, claros y concisos.	
Comentarios	
Las pruebas que generan Factura Electrónica permiten saber que los datos que se necesita de las tablas que se relacionan con este proceso son los correctos.	
Estado Aceptado/No aceptado	
Aceptado	

5.05 Especificación de pruebas de carga

Estas pruebas se las realiza en plataformas de prueba del desarrollo como, también en plataforma de producción del cliente. Con esto se busca comprobar el rendimiento del sistema en distintos escenarios.

En estas pruebas se debe tomar en cuenta los factores externos ya que pueden influir en los resultados obtenidos como la topología de red, la configuración de los servidores, etc.

Con estas pruebas se intenta medir el rendimiento de la aplicación entregada en la institución establecida.

Los tipos de pruebas de rendimiento que se utilizan más comúnmente son:

Prueba normal. Permite establecer los tiempos medios de respuesta cuando un usuario está adentro de la aplicación.

Esto nos da una pauta para realizar comparaciones posteriores cuando existe más de un usuario está adentro de la aplicación.

Prueba con número mínimo de usuarios. Estas pruebas se las realiza con el número de usuarios mínimos concurrentes establecido.

Prueba con número máximo de usuarios. Se realizan las pruebas del sistema con el número de usuarios máximo concurrentes establecido.

Prueba de número máximo soportado de usuarios. Se busca encontrar cuál es el límite del sistema.

Tabla 35

Detalle de un tipo de prueba de carga más baja

Identificador de la Prueba:	**PRCA001**
Tipo de Prueba	Prueba normal (Prueba de Carga)
Objetivo de la Prueba	Establecer los tiempos de respuesta cuando un usuario está adentro de la aplicación.
Descripción:	
Con esta prueba se busca tener una referencia futura para posteriores comparaciones así como medir unitariamente el manejo de la aplicación entregada.	
Resultados Esperados	
Que los procesos del que se realizan en el sistema sean óptimos y tengan un margen en tiempo óptimo.	
Comentarios	

Tabla 36

Detalle de un tipo de prueba de carga con un número mínimo de usuarios

Identificador de la Prueba:	PRCA002
Tipo de Prueba	**Prueba con número mínimo de usuarios**
Objetivo de la Prueba	Establecer los tiempos de respuesta cuando existen más de un usuario adentro de la aplicación.
Descripción:	
Las pruebas del sistema con el número de usuarios mínimos concurrentes establecidos en la aplicación.	
Resultados Esperados	
Validar la funcionalidad del sistema con un mínimo de usuarios logueados.	
Comentarios	

Tabla 37

Detalle de un tipo de prueba de carga con un número máximo de usuarios

Identificador de la Prueba:	PRCA003
Tipo de Prueba	**Prueba con número máximo de usuarios**
Objetivo de la Prueba	Establecer los tiempos de respuesta cuando una gran cantidad de usuarios están adentro de la aplicación.
Descripción:	
Las pruebas del sistema con el número de usuarios máximos concurrentes establecidos en la aplicación.	
Resultados Esperados	
Que los procesos del sistema sean óptimos y que mantenga tiempos de respuesta ideales cuando exista una gran cantidad de usuarios adentro de la aplicación.	
Comentarios	

Tabla 38

Detalle de un tipo de prueba de carga con un número máximo de usuarios ejecutándose hasta llegar al colapso. Se requiere descubrir los límites

Identificador de la Prueba:	**PRCA004**
Tipo de Prueba	**Prueba de número máximo soportado de usuarios**
Objetivo de la Prueba	Se busca encontrar cuál es el límite de usuarios que soporta la aplicación.
Descripción:	
Las pruebas realizadas en la aplicación con el número de usuarios máximo concurrentes establecido hasta llegar al colapso.	
Resultados Esperados	
Establecer cuál es el límite que tiene la aplicación en sobrecarga.	
Comentarios	

5.06 Configuración del Ambiente mínima/ideal

Para que exista un ambiente mínimo ideal se debe tener ciertas consideraciones como: su localización, el diseño, hardware requerido, software adecuado, conexiones de energía, temperatura adecuada, humedad mínima, contingencia contra desastres, seguridad en general. Un adecuado planeamiento permitirá que el ambiente sea el idóneo.

También se debe tomar en cuenta lo insumos básicos en las oficinas de los usuarios (toma corrientes, enchufes, espacio adecuado, conexión de red, etc.).

El ambiente en donde se encuentra situado el servidor, son los altos niveles de humedad que podrían causar condensación y electrostática. También este debe contar con detectores de humo y agua para que los equipos no sufran ningún daño, así como también UPS que protejan a los equipos de las variaciones bruscas de voltaje.

Para que esta aplicación web cuente con el ambiente mínimo para su correcto funcionamiento necesitamos:

- Un Web Hosting que brinde una conexión confiable y que permanezca siempre en línea sin desconexiones repentinas.
- El CDI bebe contar con servicio de Internet que brinde estabilidad.
- Contar con ancho de banda adecuada para que la navegación sea óptima.
- Tener condiciones físicas de las instalaciones del CDI beben se aceptables para ubicar ahí los equipos.

Capítulo VI Aspectos Administrativos

6.01 Recursos

- Instituciones Educativas (Instituto Superior Cordillera, Centro de Desarrollo Infantil Descubrir.
- Personal Administrativo del CDI (Directora del CDI, Secretaria)
- Tutor del Proyecto de Grado
- Asesores externos (Ing. Roberto Razo).
- Laptop Toshiba Core 2 Duo, 4 RAM.
- Manuales Físicos y Digitales (Instalación, Usuario, Técnico)
- Yog SQL
- NetBeans
- Rational Rose
- Power Designer
- Apache Tomcat
- Microsoft Office
- Microsoft Project
- Microsoft Visio

6.02 Presupuesto

Tabla 39

Detalle de Gastos realizados en el Proyecto (Presupuesto)

PRESUPUESTO DE GASTOS				
RUBROS	**CANTIDAD**	**UNIDAD**	**PRECIO UNITARIO**	**SUB TOTAL**
I. BIENES				
Tinta de Impresión	4	UNIDAD	$ 7,00	$ 28,00
CD´S	10	UNIDAD	$ 0,60	$ 6,00
Hojas	1500	MILLAR	$ 0,01	$ 15,00
Lapiceros	1	UNIDAD	$ 0,75	$ 0,75
II. SERVICIOS				
Movilidad				$ 80,00
Viáticos				$ 30,00
Teléfono				$ 15,00
Impresiones				$ 45,00
Fotocopias				$ 30,00
TOTAL				$249,75

6.03 Cronograma de Actividades

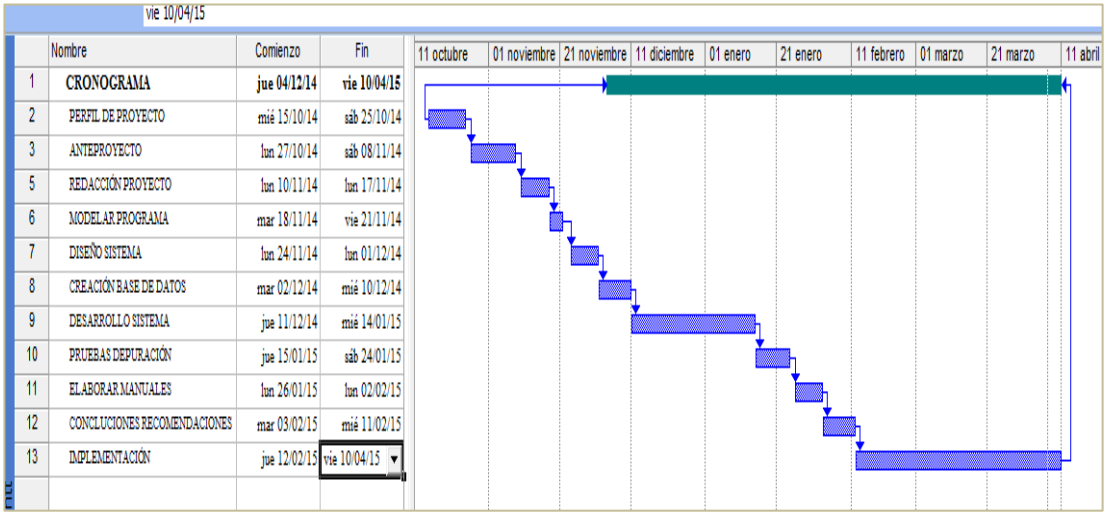


Figura 21 **Tiempos propuestos para el desarrollo y la implementación del Sistema**

Capítulo VII: Conclusiones y Recomendaciones

7.01 Conclusiones

- Al realizar el proyecto de grado con el cual permitió el desarrollo de una aplicación web de facturación electrónica en el que se puso en práctica todos los conocimientos adquiridos en los años de estudio.

- Se cumplió con todos los requerimientos que exigía el centro de Desarrollo Infantil Descubrir y con el requerimiento del Instituto Superior Cordillera.

- La aplicación web es beneficiosa para el CDI ya que al tener la facilidad de poder ingresar desde cualquier lugar tan solo contando con el servicio de internet y una laptop para realizar cualquier consulta o proceso que se requiera de parte de los usuarios de la institución

- Se diseñó una interface amigable e interactiva de acuerdo a los requerimientos indicados por los beneficiarios. Si bien es cierto muchas de las configuraciones han requerido de mucho tiempo, pero nos ha dado buen resultado ya que por medio de esto conocemos nuevas técnicas de desarrollo.

- Se pudo comprobar que el SRI se enfoca en validar los datos de la factura pero no así los rubros de la misma ya que si se envía con otro valor de igual manera la valida.

7.02 Recomendaciones

- La lectura del manual de usuario permitirá el óptimo aprovechamiento de la aplicación.

- Ingresar de manera correcta los datos en los respectivos formularios para que no exista ningún inconveniente al momento de la validación por el SRI

- Ubicar a una persona competente en el área informática como administrador del sistema para que sea capaz de resolver cualquier inconveniente de Hardware y Software.

- Realizar capacitaciones periódicas a los usuarios para el correcto manejo de la aplicación.

- Realizar actualizaciones necesarias para que la aplicación funcione de forma óptima.

- Motivar a los encargados del proceso de facturación electrónica para el uso de la aplicación ya que además de facilitar el trabajo permite cumplir con la normativa dispuesta por el SRI.

- Realizar charlas para que los miembros del CDI conozcan de nuevas tecnologías que facilitan y mejoran su ambiente de trabajo y potencian su desempeño laboral.

ANEXOS

A.01 Entrevista

DISEÑO ENTREVISTA		
identificador: 001		
Preguntas	**Objetivos**	**Análisis posterior**
¿Cómo se realiza el proceso de facturación en la actualidad?	Obtener información de la manera como se lleva a cabo el proceso de facturación.	Se debe contar con un registro de alumnos Contar con el registro de representante financiero que realiza el pago de pensiones y matricula. Emisión de factura de cada estudiante. Validación de la información para guardar en la factura electrónica. Envió de Factura al correo electrónico del cliente.
¿Cuál es la problemática que buscan solucionar a través del sistema web de facturación electrónica?	Determinar los problemas que presenta la institución para mejorar con la implementación del sistema web de facturación electrónica.	Se quiere tener reportes de la información de los alumnos que se realizó los pagos y de cuantos faltan por pagar. La Directora necesita tener un control de la información de los pagos de cada estudiante ya esto permitirá cumplir con la ley vigente.
¿Quiénes necesitan tener acceso al sistema de facturación?	Obtener el listado de los usuarios que manejarán el sistema de facturación.	Las personas que tendrán el acceso a la al sistema: Directora, Secretaria
¿Con qué aplicaciones trabaja en la actualidad?	Aplicaciones que utilizan para su trabajo diario.	No se utiliza ninguna aplicación informática. Todo se registra en un libro con datos personales del alumno.
¿Cuáles son las dificultades que se presentan más frecuentemente al realizar este proceso?	Evaluar cuáles son las dificultades que se presentan con cada uno de los involucrados.	La señora Directora y su secretaria necesitan tener la información actual y confiable de cada alumno. La secretaria necesita entregar información de las facturas con datos correctos a los padres de familia.
¿Qué información de los representantes de los alumnos es necesaria para llenar la factura?	Conocer que información se requiere de parte del representante para llenar la factura.	Cédula o RUC del representante. Nombres y apellidos. Nombres y apellido del alumno de quien se realiza el pago. Dirección del representante o cliente. Teléfono del representante o cliente. Fecha del pago que de la factura.

		Datos del estudiante. Correo electrónico
¿Con qué información del CDI debe contar la cabecera de la factura?	Conocer que información del CDI es necesaria para que la factura tenga validez.	RUC del CDI. Nombre de la institución. Razón social. Dirección del CDI. Teléfono del CDI. Correo electrónico del CDI. Autorización de factura del CDI. Número de Factura del CDI.
¿Con qué información debe tener el detalle de la factura?	Conocer que información necesaria para que al llenar la factura esté correcta.	Valor a pagar por alumno en la factura. Descripción del pago es decir de qué alumno se realizó el pago. Quien lleno la factura. Descuento en caso de haberlo. IVA tarifa 0% por ser servicios educativos. Valor total de la factura.

A.02 Detalle de requerimientos

Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
	REQUERIMIENTOS FUNCIONALES					
RF001	La Directora y o el administrador del sistema de facturación del CDI necesitan llenar el formulario de registro del alumnos y representantes.	Directora	Media	Sistema	En revisión	Directora Administrador
RF002	La Directora y o el administrador del sistema del CDI necesitan llenar el formulario del representante financiero si los estudiantes son apadrinados.	Directora	Media	Sistema	En revisión	Directora Administrador
RF003	El administrador del sistema del CDI necesita llenar el formulario de información de la institución.	Directora	Alta	Sistema	En revisión	Directora Administrador
RF004	Generar factura electrónica con los datos de institución, representante financiero y de alumno.	Directora	Alta	Sistema	En revisión	Directora Secretaria
RF005	Firmar electrónicamente, guardar y enviar el archivo XML para que el SRI lo valide.	Directora	Alta	Sistema	En revisión	Directora

REQUERIMIENTOS NO FUNCIONALES						
RNF001	La Directora y la Secretaria quieren ingresar desde cualquier lugar a revisar los pagos realizados de matrícula o pensiones.	Directora	Media	Usuario	En revisión	Directora Secretaria
RNF002	Se requiere que las personas que tengan acceso al sistema de facturación sean:	Directora	Media	Usuario	En revisión	Directora Secretaria
RNF003	La Directora o Secretaria necesitan enviar la factura electrónica ya validada a los correos electrónicos de los representantes financieros de los estudiantes.	Directora	Media	Usuario	En revisión	Directora Secretaria

A.03 Detalle interés, involucrados y conflictos percibidos del proyecto

Actores Involucrados	Intereses sobre el problema central	Problemas percibidos	Recursos, Mandatos y Capacidades	Intereses sobre el Proyecto	Conflictos Potenciales
Directora del Centro de Desarrollo Infantil	Necesidad de contar con reportes rápidos de toda la información de los pagos de facturas del centro educativo.	Los reportes se tardan en elaborar porque se los realiza manualmente.	Necesita mejorar los tiempos de respuesta para tener un manejo más adecuado de la información requerida	Tiene un significativo por la elaboración de la aplicación, la que agilitará notablemente el proceso.	Con la secretaria que tiene resistencia al cabio del manejo de la información manual Con padres de familia ya que algunos no están familiarizados con la tecnología.
Secretaria	La secretaria necesita entregar información consistente a los representantes de los alumnos.	La duplicidad de la información genera errores en la emisión de las facturas.	Requiere entregar los comprobantes de pagos con la información sin errores.	Es de gran interés por que el tiempo que se invertía en el llenado manual de las facturas se reducirá y los padres de familia tendrán un mejor servicio.	
Servicio de Rentas Internas	Al realizar la validación de la información que se genera en la institución tiene un respaldo que a la institución cobre los rubros correctos si perjudicar ni al estado tampoco al cetro de	La poca información que tienen las instituciones educativas sobre las leyes que rigen en el país.	Saber que esa información será entregada directamente a sus padres.	Gran interés en la implementación de la aplicación, considerando que la información generada en la institución educativa tendrá un respaldo de la entidad reguladora.	Con la Directora ya que tiene un mínimo conocimiento de las nuevas normativas que rigen actualmente sobre facturación electrónica.

	desarrollo infantil.				
Alumnos	Tendrán toda su información organizada y a la mano para cualquier consulta o requerimiento.	Las notificaciones de pagos no realizados, que se envían en notas a los estudiantes para que revisen sus padres.	Necesitan ágilmente consultar notas	Los niños contaran con la información necesaria para ellos sin involucrarlos en el proceso de manera directa.	
Padres de familia	Recibirán la factura electrónica en su correo electrónico después de realizar el pago de matrícula o pensiones del año electivo.	Las facturas se entregan de forma física la misma que se puede extraviar o, al manipularla se deteriora o se rompe.	Necesitan las facturas sin tachones ni enmendaduras para realizar declaraciones de impuestos u otros trámites.	Los padres recibirán sus facturas de manera más sencilla mejorando en la parte de la recepción del documento al correo electrónico.	Con el proceso ya que no tienen un conocimiento profundo de la forma que se lleva la facturación electrónica.

A.04 Manual de Instalación

Manual de Instalación

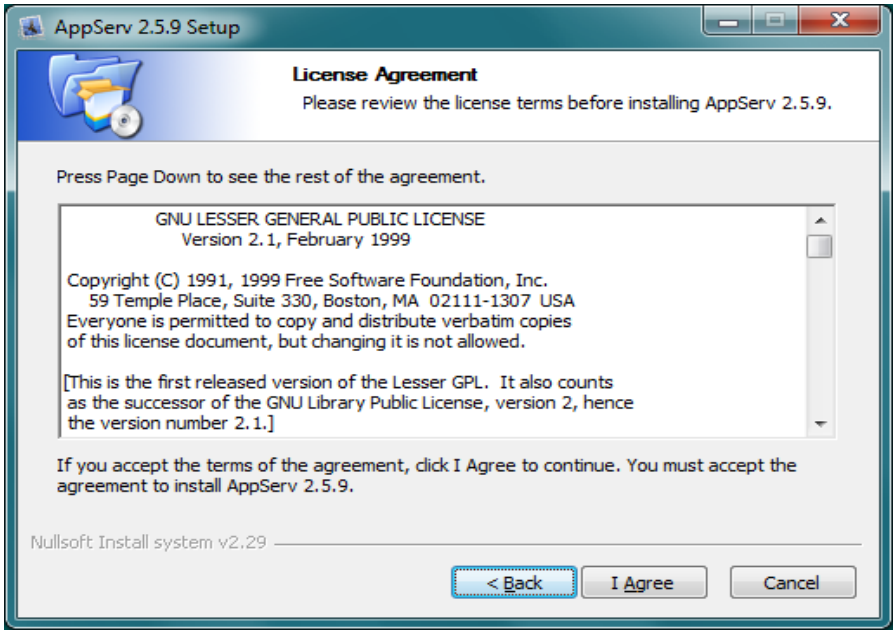
INDICE

Titulo	Página
1. Instalación de MySQL	73
2. Instalación de SQLYOG	77
3. Configurar SQLYOG para conectar a MYSQL	80
4. Instalación de Apache Tomcat	82
5.- Ejecutar la aplicación	85

1. Empezamos la Instalación de MySQL

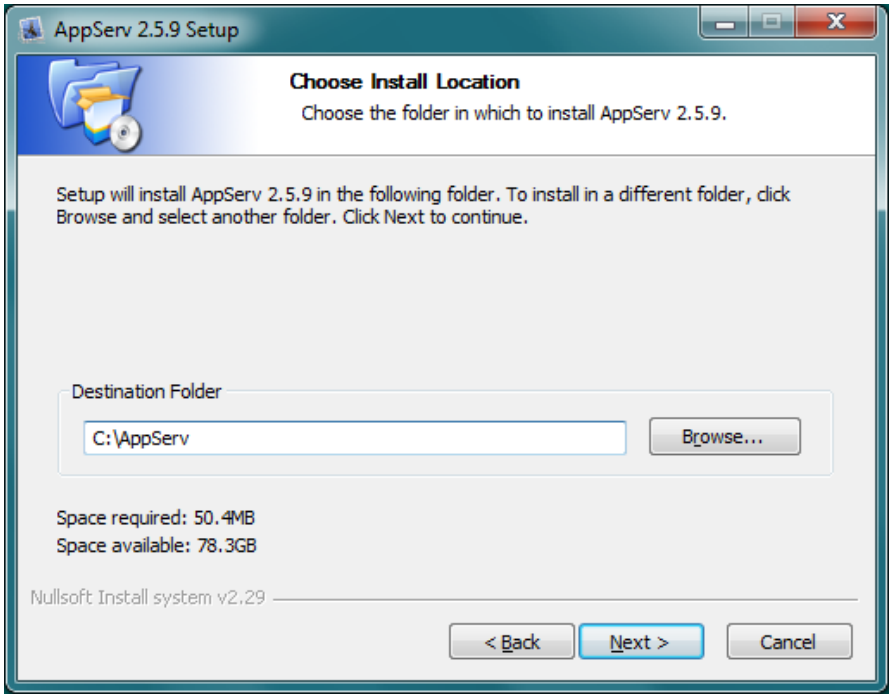
1.1. Damos 2 clics al archivo de instalación appserv-win32-2.5.9.exe. En la ventana que aparece damos clic en el botón Next o Continuar.

1.2. En aceptar los términos de licencia damos clic en el botón I Agree.

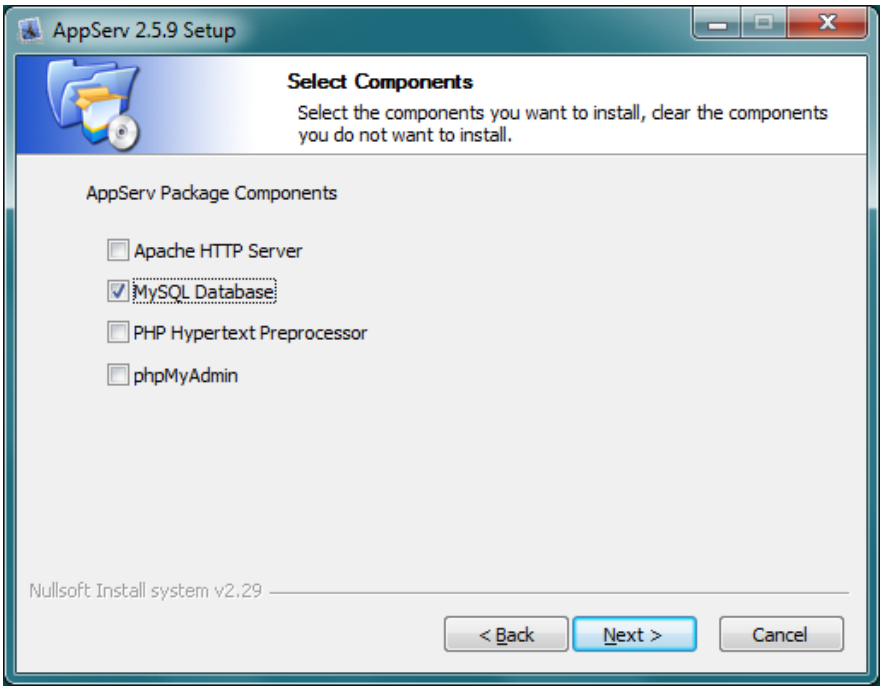


1.3. Damos clic en Browse para especificar donde se va a guardar el archivo.

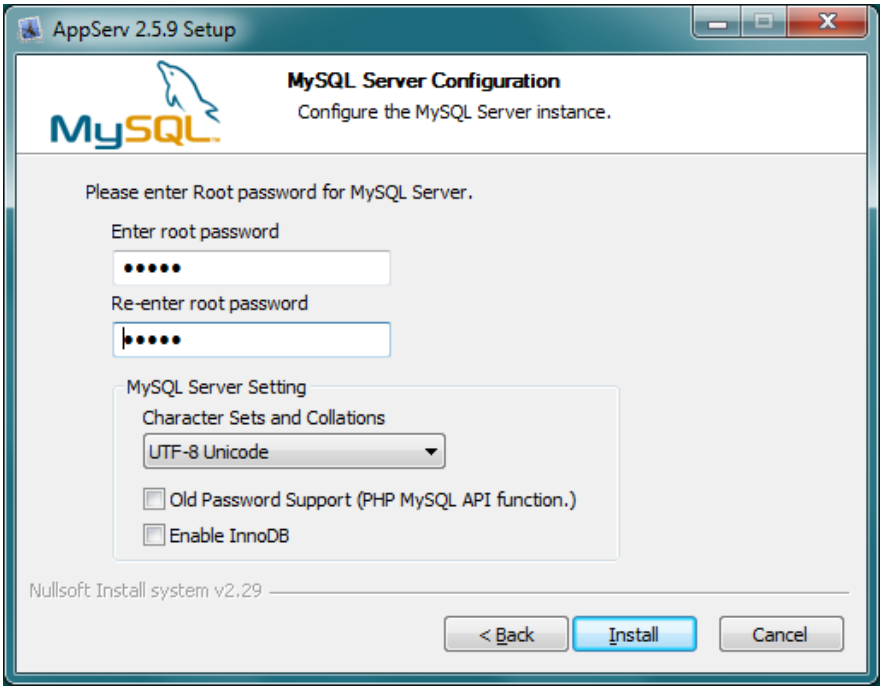
Dejar esta opción por defecto C:\AppServ y damos clic en Next.



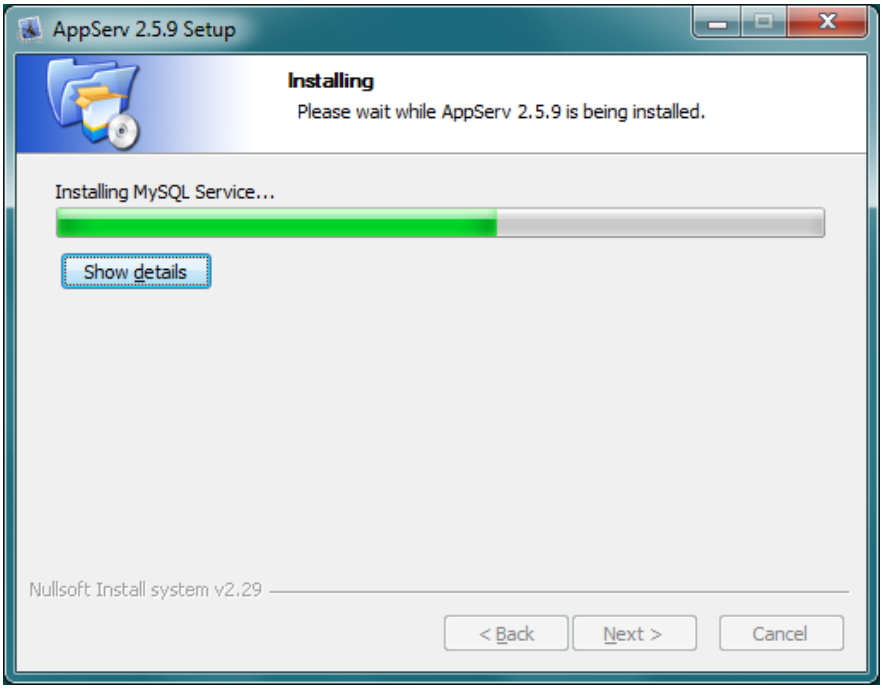
1.4. Seleccionamos el paquete MySQL Database y damos clic en Next.



1.5. Debemos ingresar una contraseña para root, en este caso le hemos puesto root y confirmamos la contraseña, luego damos clic en Install.



1.6. El programa comienza a instalarse.

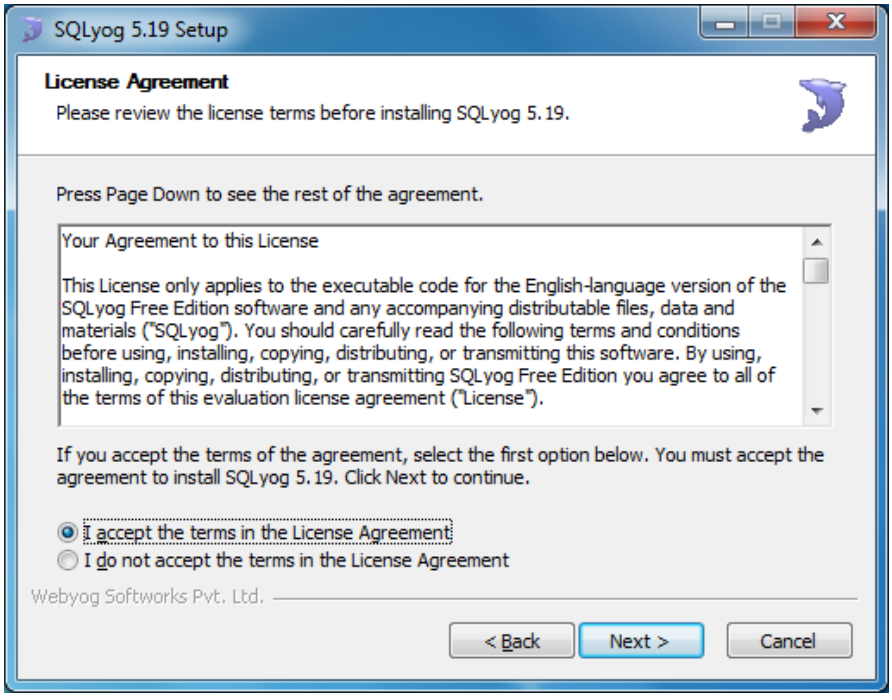


1.7. Seleccionamos Start MySQL para iniciar el servicio y damos clic en Finish.

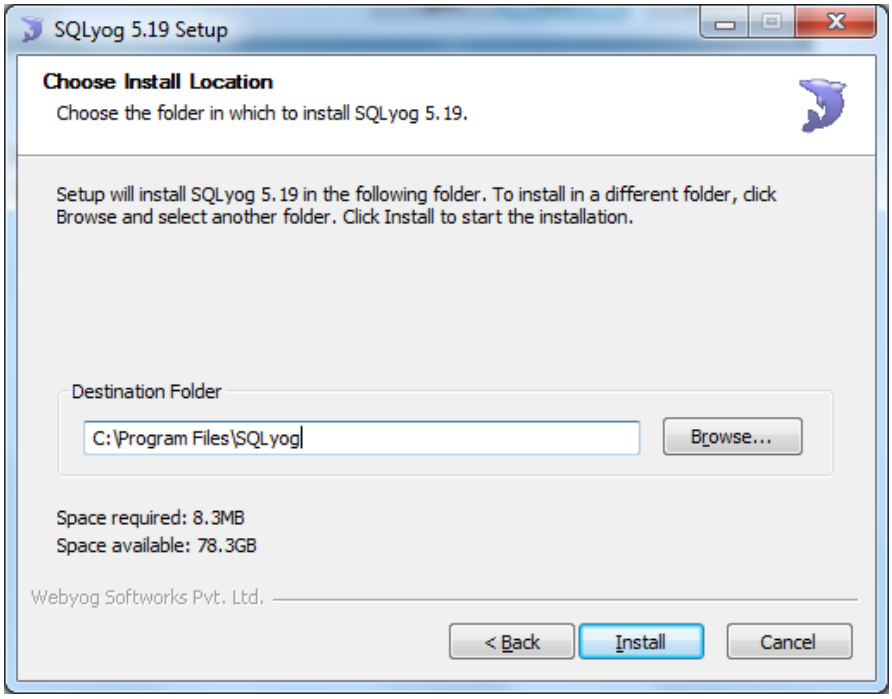
2. Instalación de SQLYOG

2.1. Damos doble clic al archivo del instalador SQLyog519.exe. En la ventana que nos aparece damos clic en el boton Next.

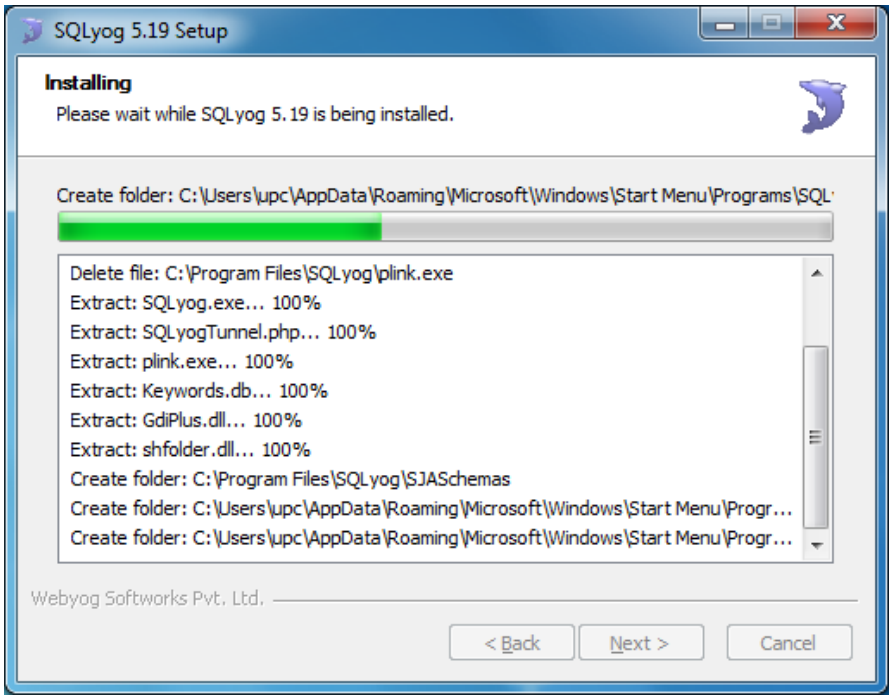
2.2. Seleccionamos en aceptar los términos de licencia y damos clic en Next.



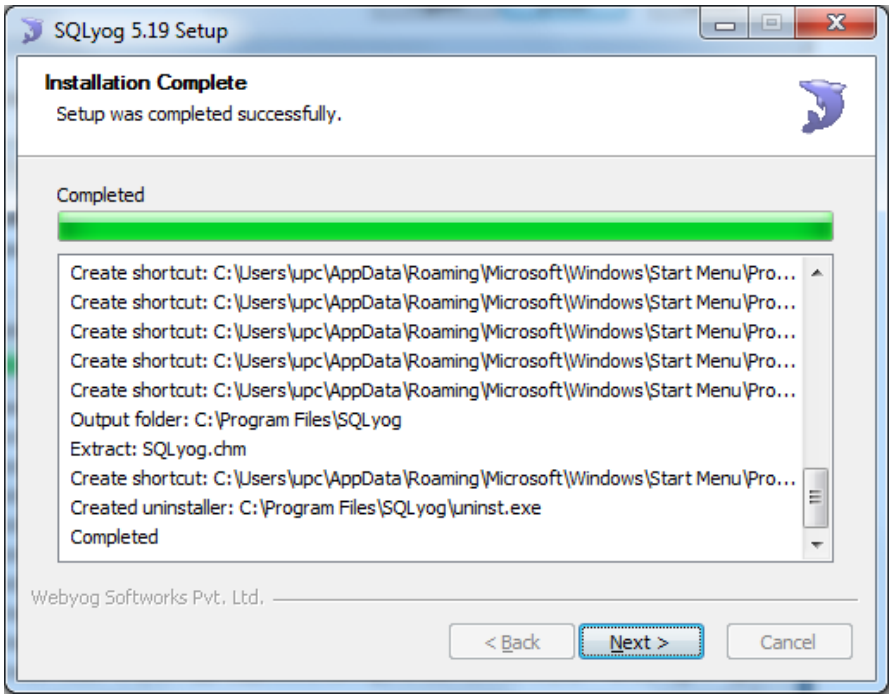
2.3. Damos la ruta donde se instala el programa. Dejamos la ruta por defecto del programa C:\Program Files\SQLyog y damos clic en Install.



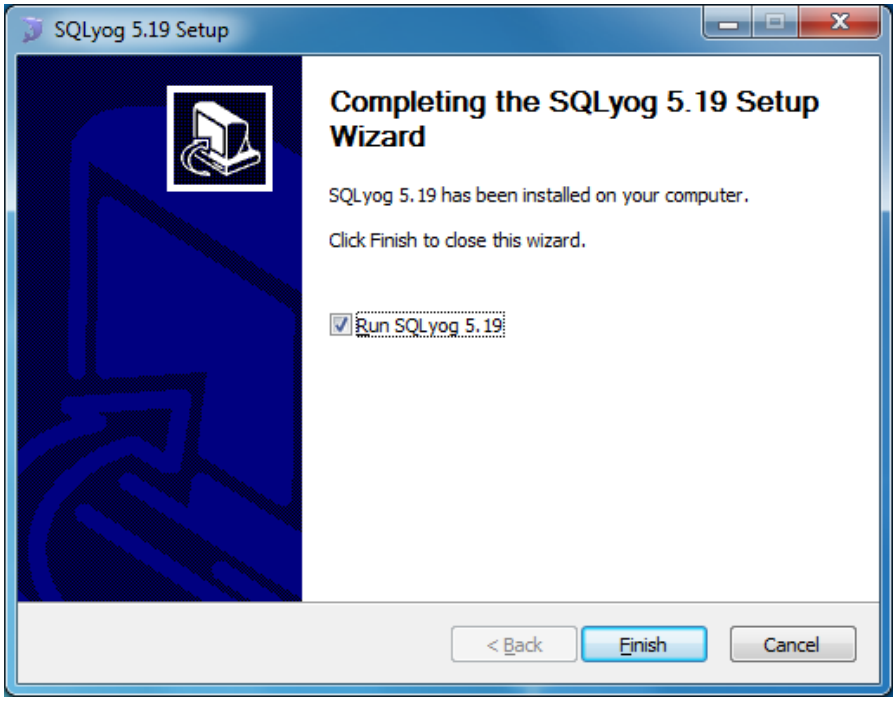
2.4. El programa empieza a instalarse.



2.5. Una vez nos muestre completa la instalación damos clic en el botón Next.

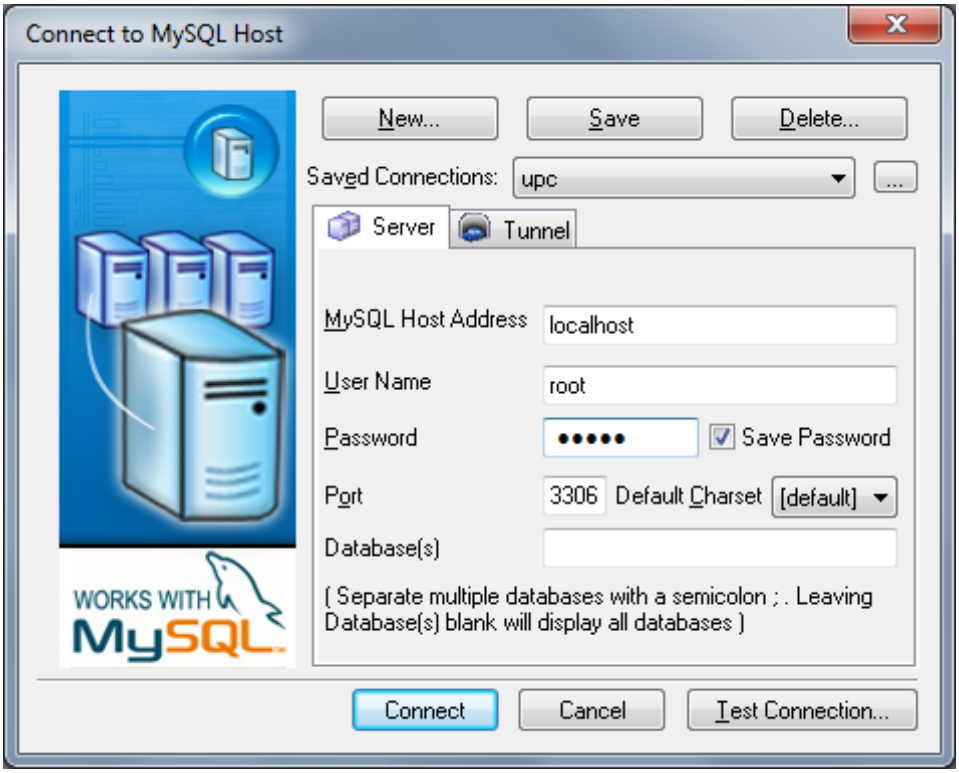


2.6. Damos clic en el botón Finish.

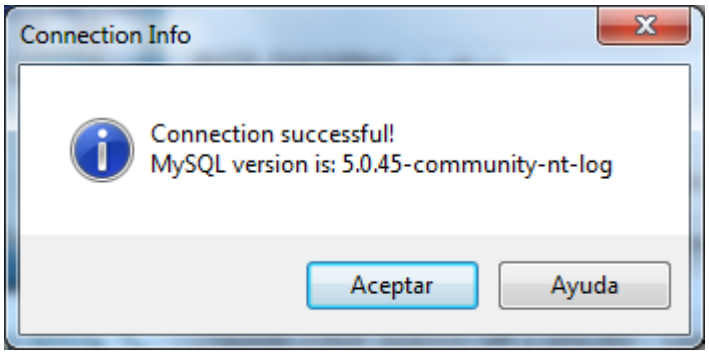


3. Configurar SQLYOG para conectar a MYSQL

3.1. En la venta de conexión nos pide el password de root la cual es root. Luego damos clic en el botón Test Connection.

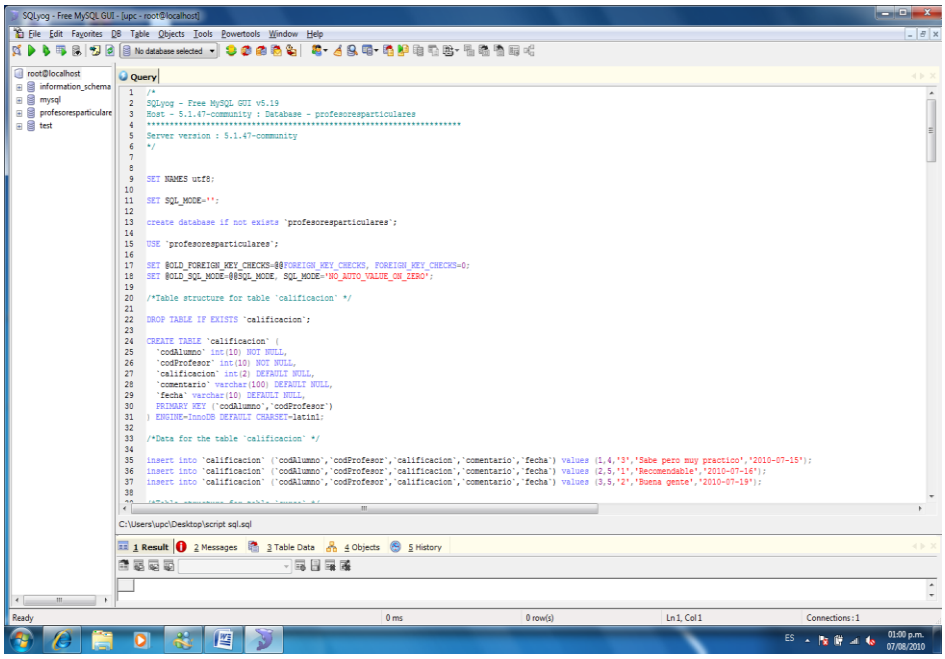


3.2. Aparece la conexión es correcta. Si sale mensaje de error debemos revisar los pasos anteriores.

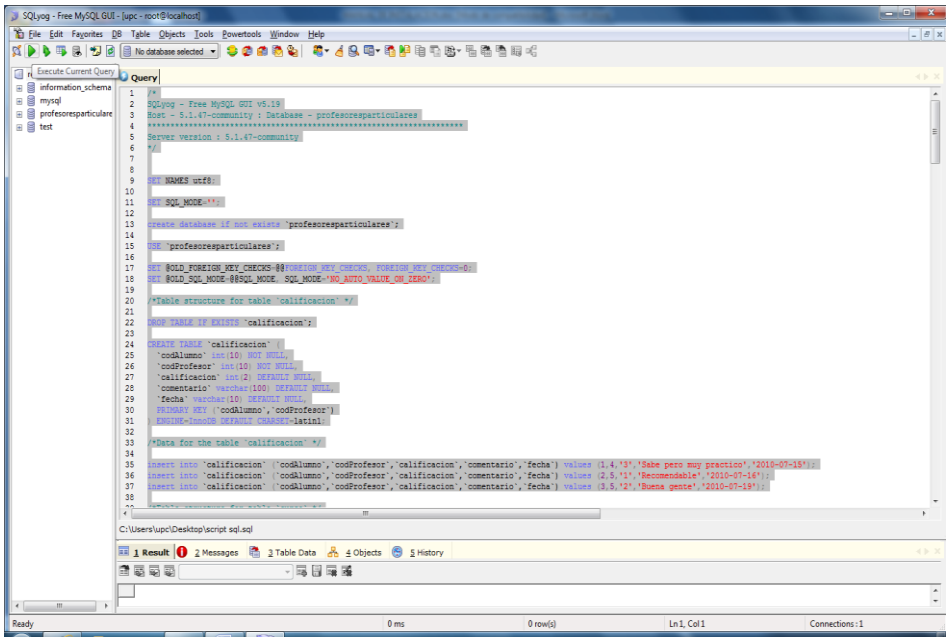


3.3. Para entrar a la base de datos damos clic en el botón Connect.

3.4. Desde la interfaz gráfica abrimos el archivo script: aca_factura.sql.



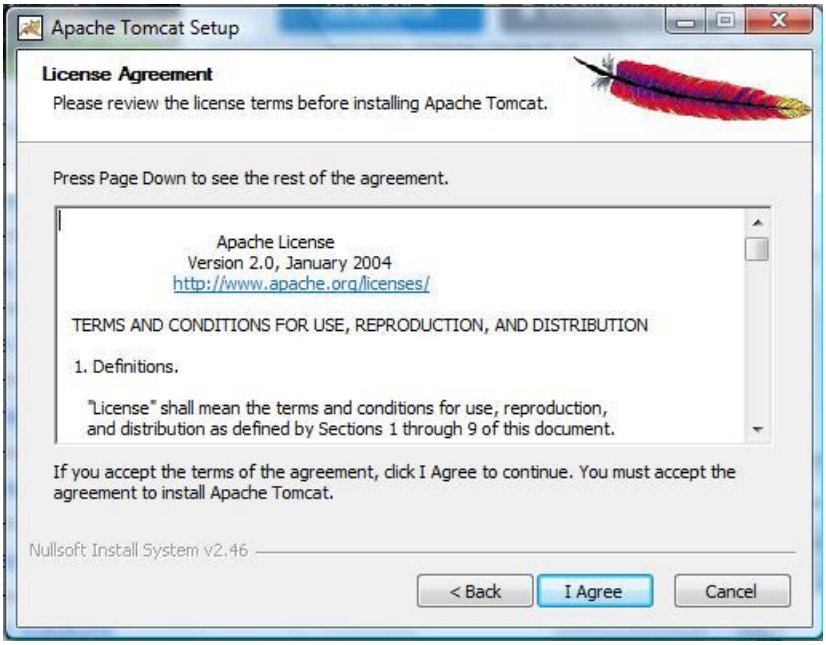
3.5. Para finalizar, corremos el script, seleccionamos todo y damos clic en el botón Execute Current Query.



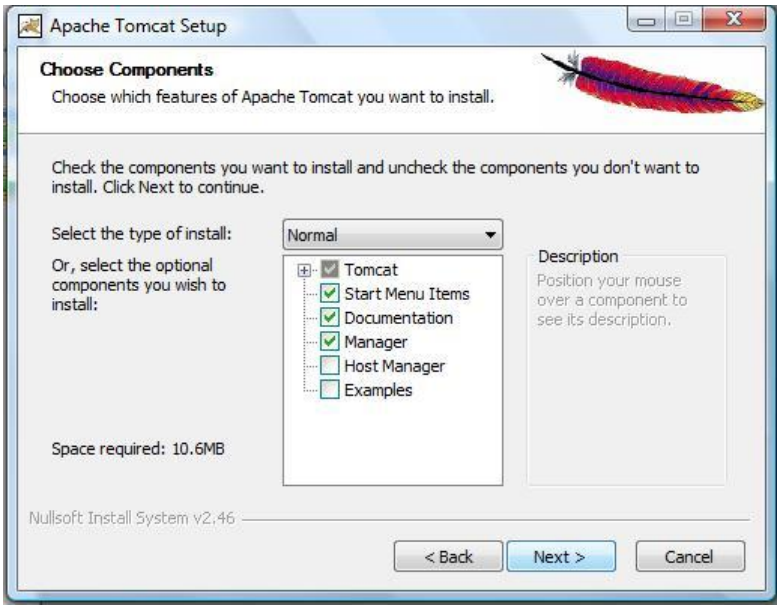
4. Instalación de Apache Tomcat.

4.1. Damos doble clic en el ejecutable de la instalación y nos aparece el asistente de instalación, damos clic en el botón Next>

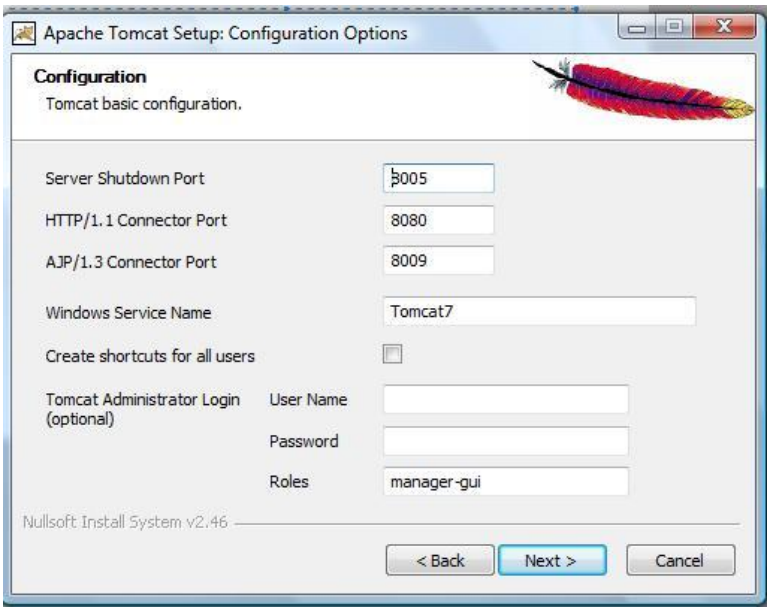
4.2. En la pantalla del asistente damos clic en I agre ósea acepto los términos de la licencia.



4.3. En los componentes que deseamos que se instalen dejamos los que están predominados y damos clic en Next >



5.4. En la configuración dejamos también los predeterminados y damos clic en Next >.



5.5 Aceptamos y ya lo tenemos lito al servido.

5. Ejecutar aplicación.

5.1. Hacer un SVN Checkout y en URL of repository poner la siguiente dirección

5.1.1. http://localhost:8080/aca_factura/login.xhtml y darle clic en OK

5.2. Iniciar desde el netbeans el tomcat, luego abrir el proyecto descargado.

5.3. Seleccionamos el proyecto, damos clic derecho y seleccionar la opción Clean and Build. Luego si todo hasta que salga ok, damos clic en el botón RUN.

5.4. La aplicación empieza a ejecutarse.

A.05 Manual de Usuario

Manual de Usuario

INDICE

Titulo	Página
1. Ingreso del Administrador del sistema	87
2.- Ingreso al sistema	87
3.- Ingresos de Usuarios	88
4.- Menú general del sistema	88
5.- Ingreso a Registros de usuarios del sistema	89
6.- Ingreso de periodo al sistema	90
7.- Ingreso de cursos al sistema	91
8.- Ingreso de rubros al sistema	92
9.- Ingreso de rubros por mes	93
10.- Ingreso a gestión	94
11.- Ingreso a Registro de estudiantes	95
12.- Ingreso a facturación	96
13.- Ingreso a Facturación electrónica	97
14- Ingreso a cambio de password del usuario	98
15.- Salida del sistema	99

1.- Ingreso del Administrador del sistema

El administrador genera su usuario y contraseña desde la base de datos, el cual podrá asignar roles a, claves de usuario y password a cada persona que tenga acceso a la aplicación.

2.- Ingreso al sistema

Los usuarios ingresan por el navegador de su computador al dominio o dirección electrónica en donde se encuentra la aplicación.

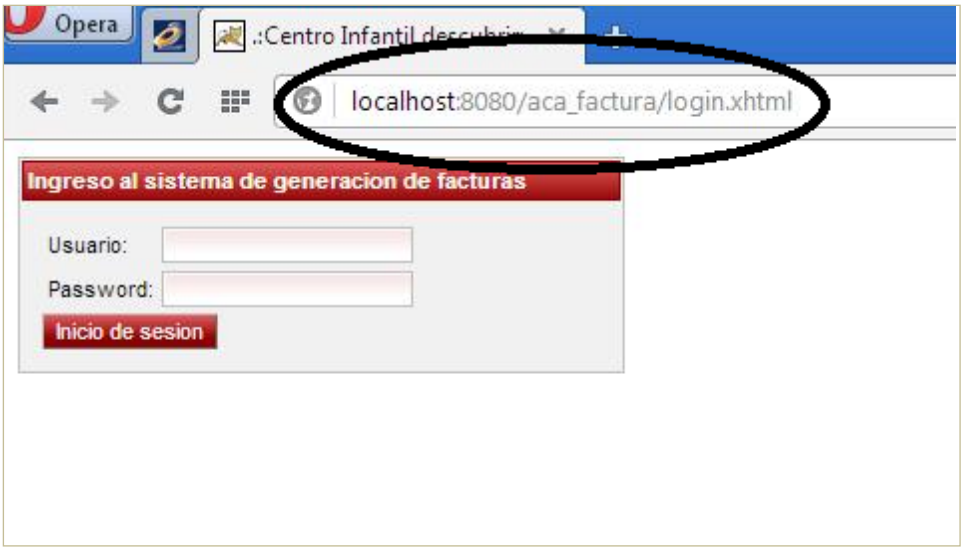


Fig.1 Ingreso al sistema de Facturación

3.- Ingresos de Usuarios

Con nuestro usuario y password la misma que consta de por lo menos de 6 caracteres que pueden ser números o letras y el usuario que puede ser el nombre del usuario, si olvido su contraseña el usuario debe pedir al administrador que le asigne otra.

Fig.2 Ingreso de usuarios al sistema

4.- Menú general del sistema

En el menú tenemos 4 opciones la primera que es la pestaña desplegable de parámetros, la segunda pestaña desplegable de gestión, la tercera que es la pestaña de cambiar password y la última que es la pestaña de salir.

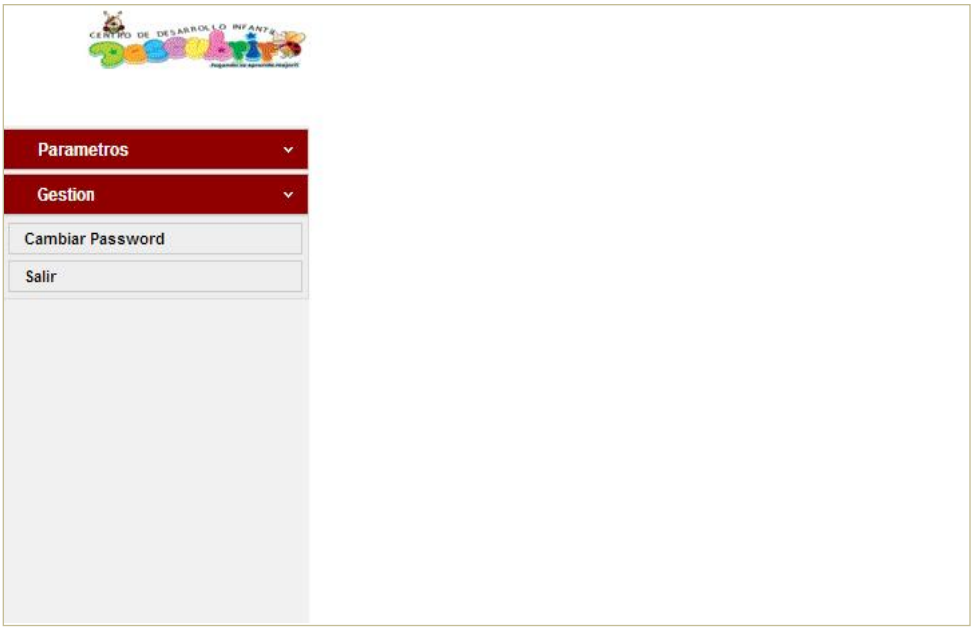


Fig.3 Menú general del sistema

5.- Ingreso a Registros de usuarios del sistema

Damos clic en parámetros y nos muestra la primera pestaña del formulario de registro de usuario en donde ingresamos los datos personales de cada usuario con su respectivo código, clave también aquí le asignamos los respectivos perfiles y estado.

Se puede realizar búsqueda de los usuarios por su nombre realizarle cambios en su información o eliminarlo por completo dependiendo la necesidad del administrador del sistema.

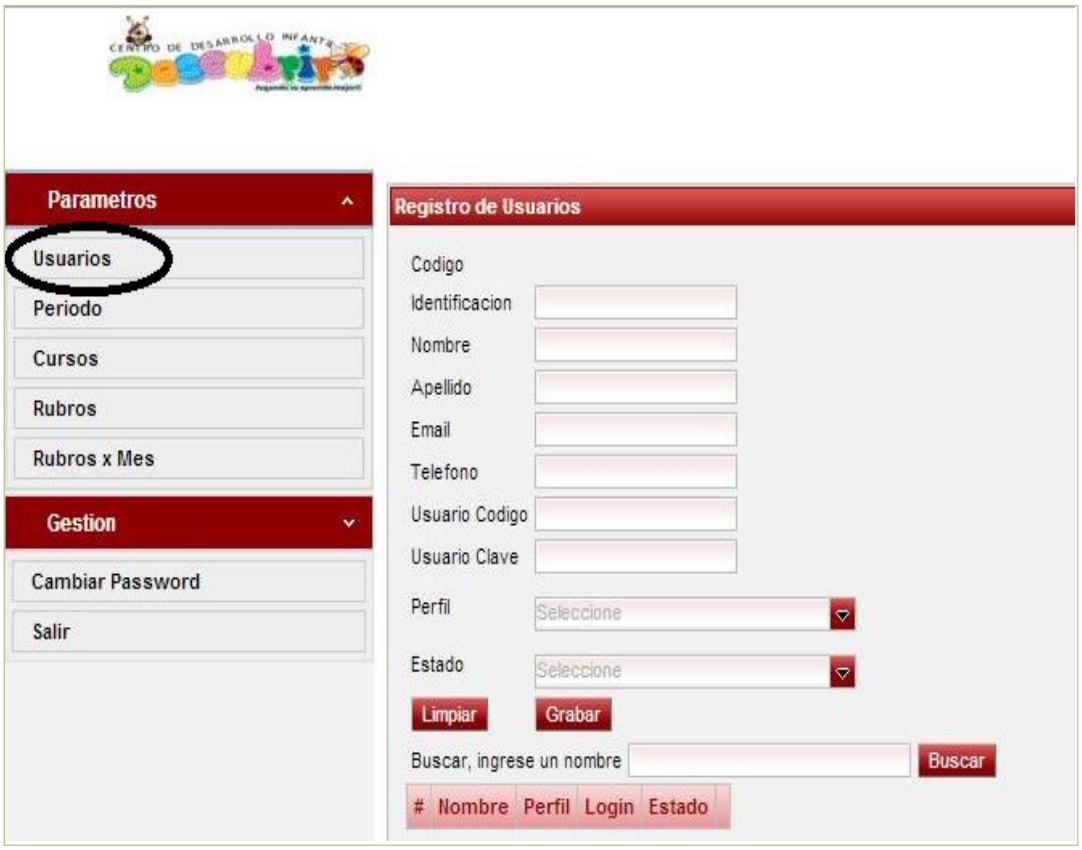


Fig.4 Registro de usuarios al sistema

6.- Ingreso de periodo al sistema

Damos clic en la pestaña periodo y nos muestra el formulario que dice periodo escolar en donde llenamos los diferentes campos como año en donde ubicaremos el año actual sección el cual llenaremos con la información de distribución de los alumnos por sus edades, jornada si es: media jornada, media jornada con el almuerzo y jornada completa.

Fig.5 Ingreso de Pariodo al sistema

7.- Ingreso de cursos al sistema

Damos clic en la pestaña curso en donde se muestra el formulario curso el mismo que se llena con la información que se encuentra precargada en la pestaña desplegable y de igual manera el paralelo, también me permite modificar o eliminar cursos seleccionándolos de la grilla que se muestra en la parte inferior del formulario.

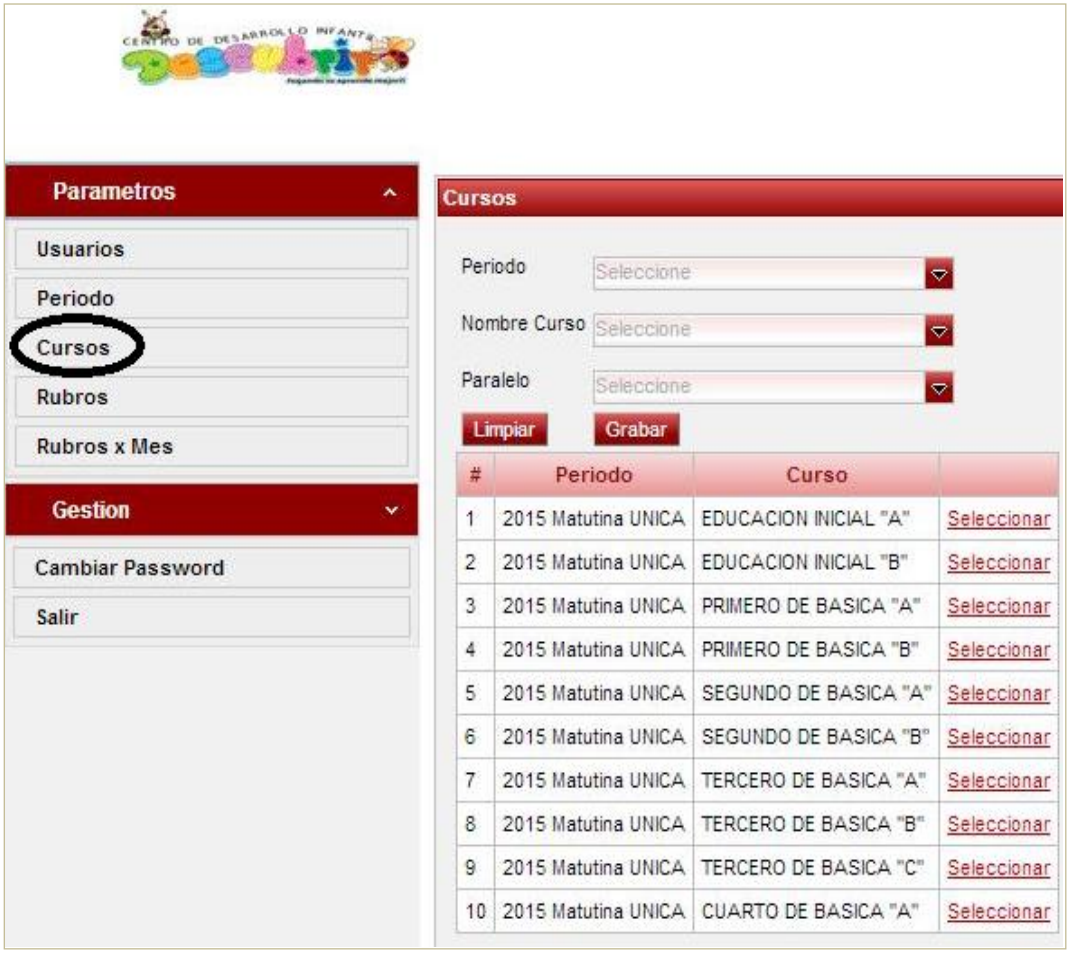


Fig.6 Inge4so de usuarios al sistema

8.- Ingreso de rubros al sistema

Damos clic en la pestaña rubro y nos muestra el formulario de registro de items el cual llenamos con datos de que estamos pagando de pensión o matricula si necesitáramos cobrar el IVA también se lo puede seleccionar para que nos calcule.

Fig.7 Ingreso de rubros al sistema

9.- Ingreso de rubros por mes

Damos clic en la pestaña de Rubros por mes y nos muestra el formulario que dice ítems curso en el cual podemos ingresar desde la pestaña de selección el curso y el ítem el cual nos indica de que vamos a facturar, seleccionamos el mes, el valor y la fecha de cobro, también se puede modificar o eliminar los valores que se cargan a ese mes.

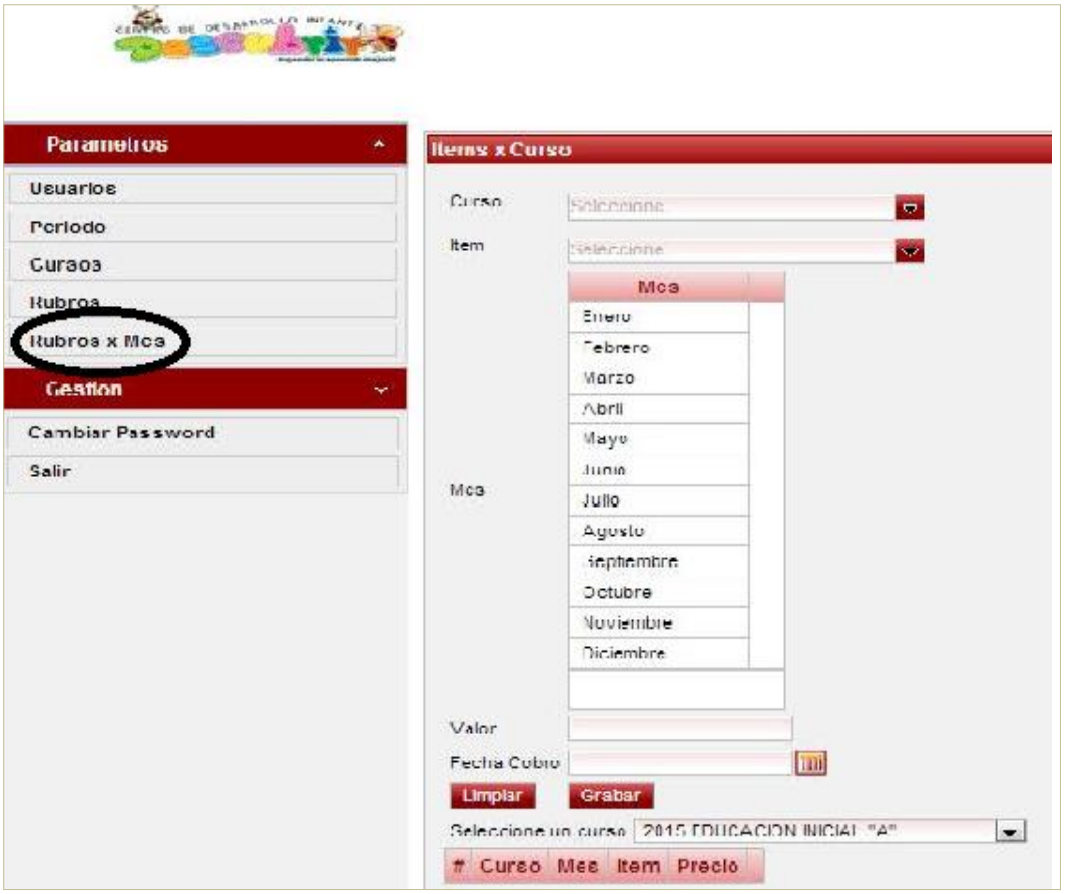


Fig.8 Ingreso de rubros al sistema

10.- Ingreso a gestión

En el menú tenemos 4 opciones la segunda que es la pestaña desplegable de gestión, que nos muestra las pestañas de estudiantes, cargar banco, facturación ventanilla, factura electrónica, se puede seleccionar cualquiera de ellas.

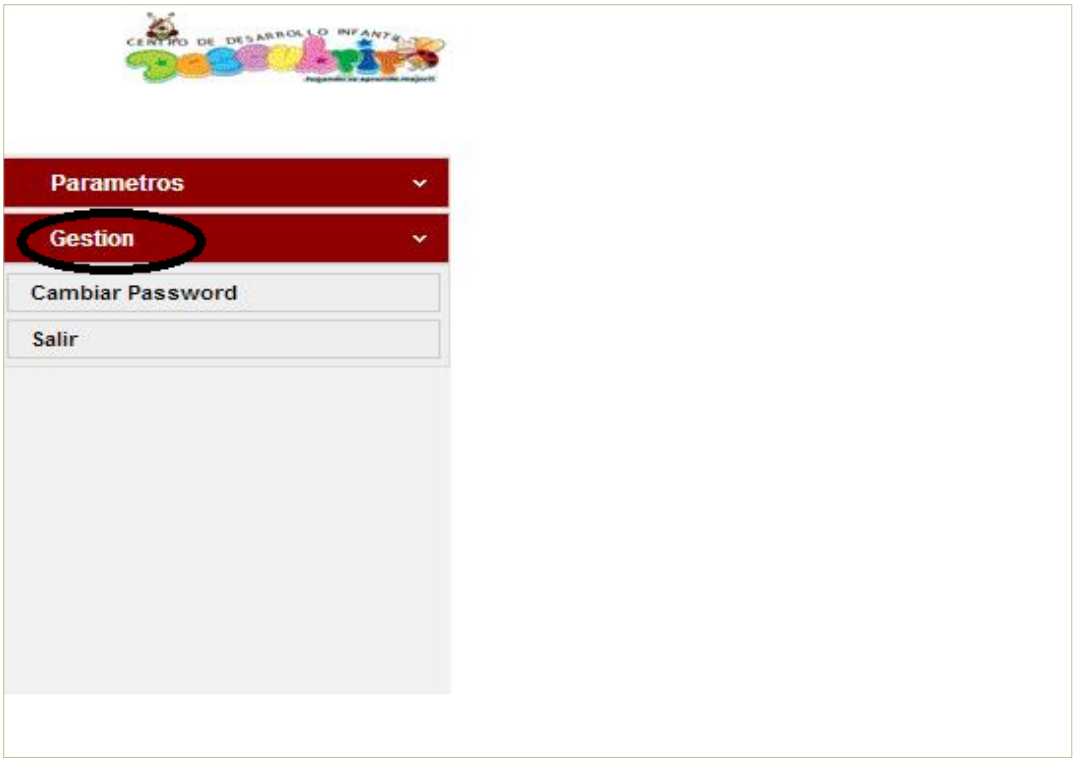


Fig.9 Ingreso a gestión del sistema de facturación electrónica

11.- Ingreso a Registro de estudiantes

Damos clic en la pestaña curso en donde se muestra el formulario de registro de estudiantes por curso mismo que se llena con la información que se encuentra precargada en la pestaña desplegable de curso, tipo de identificación si es cedula o pasaporte y los datos de cada estudiante, en la parte de datos para facturación seleccionamos el tipo de identificación del representante y sus datos generas, también si cuenta con beca de cuanto es su valor.

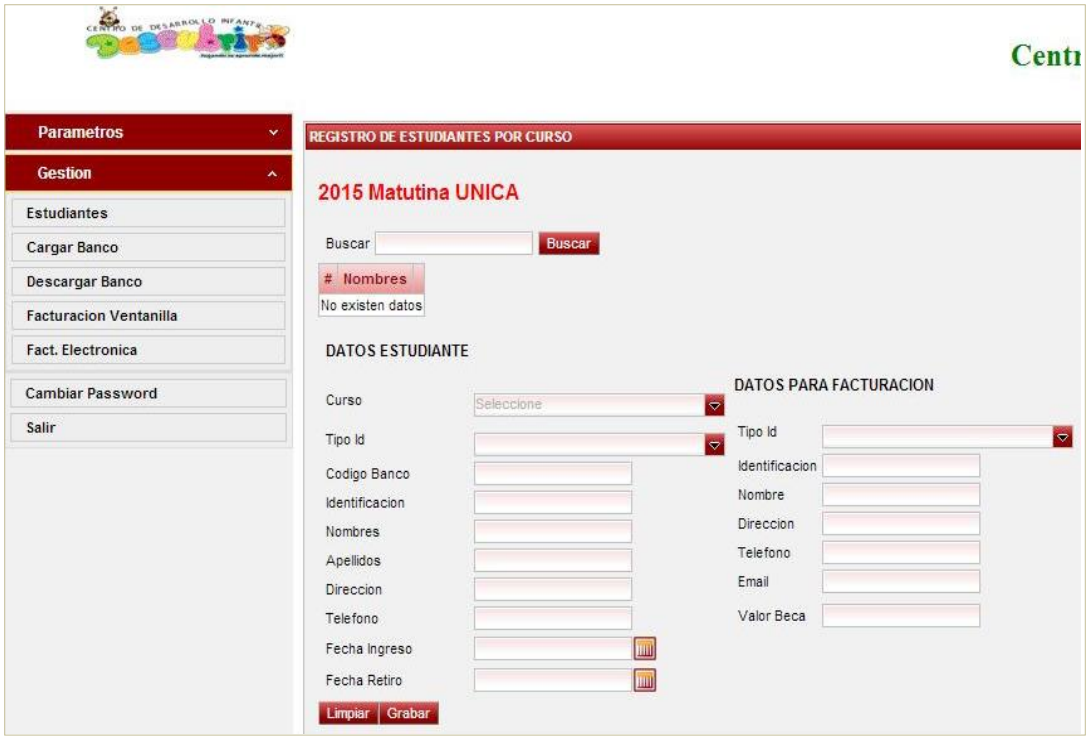


Fig.10 Ingreso al registro de cada estudiante

12.- Ingreso a facturación

Damos clic en el menú facturación por ventanilla y nos permite cargar los datos para generar una factura física la cual la podemos imprimir pero que serviría solo como comprobante de pago ya que no contaría con ninguna tributaria.

Fig.11 Ingreso a facturación por ventanilla

13.- Ingreso a Facturación electrónica

Damos clic en la pestaña de facturación electrónica y nos muestra el formulario de facturación por lote la cual nos permite generar las facturas y su XML para enviarlas al SRI para que las validen y cuenten con la información para que sea un documento electrónico valido.

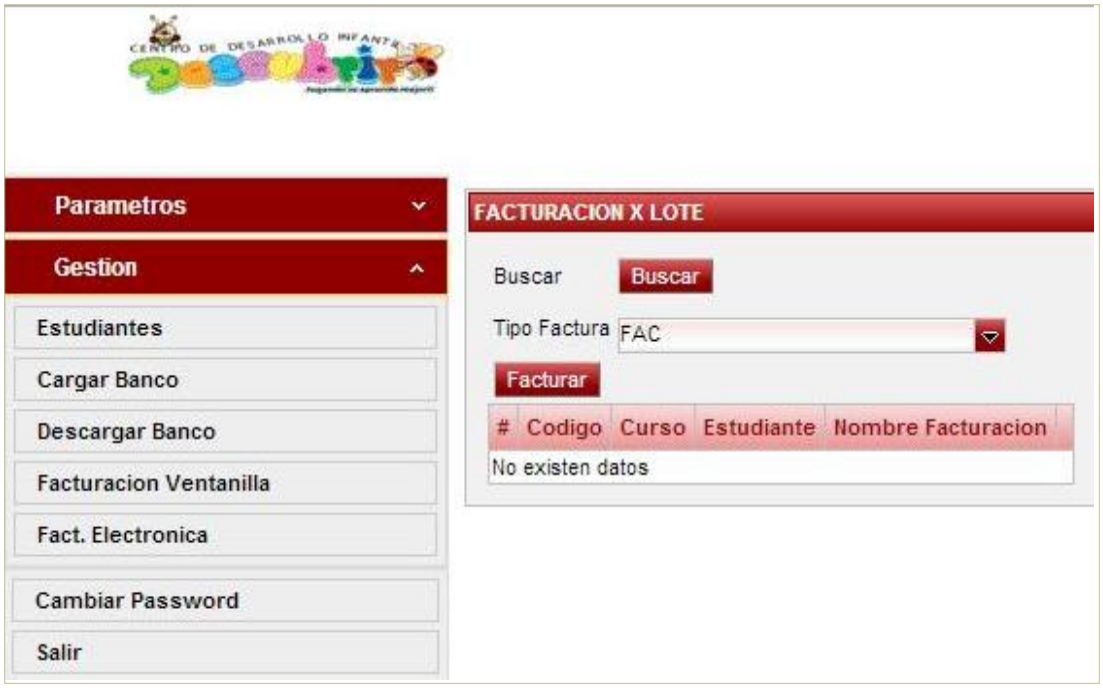


Fig.12 Ingreso a facturación electrónica del sistema

14- Ingreso a cambio de password del usuario

Damos clic en la pestaña de cambiar password y nos muestra el formulario que nos permite cambiar la password ingresando nuestro código y la nueva clave.

Fig.13 Cambio de password de usuarios

15.- Salida del sistema

Damos clic en la pestaña de salir y nos permite cerrar nuestra sesión y nos lleva al formulario de ingreso al sistema.

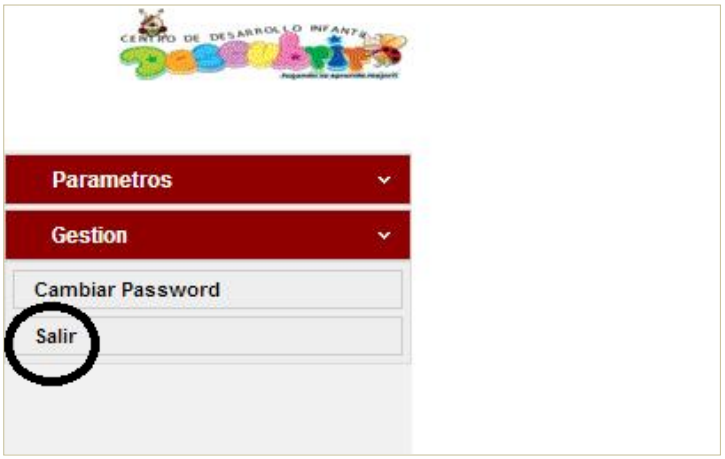


Fig.14 Salida del sistema de facturación electrónica

A.06 Manual Técnico

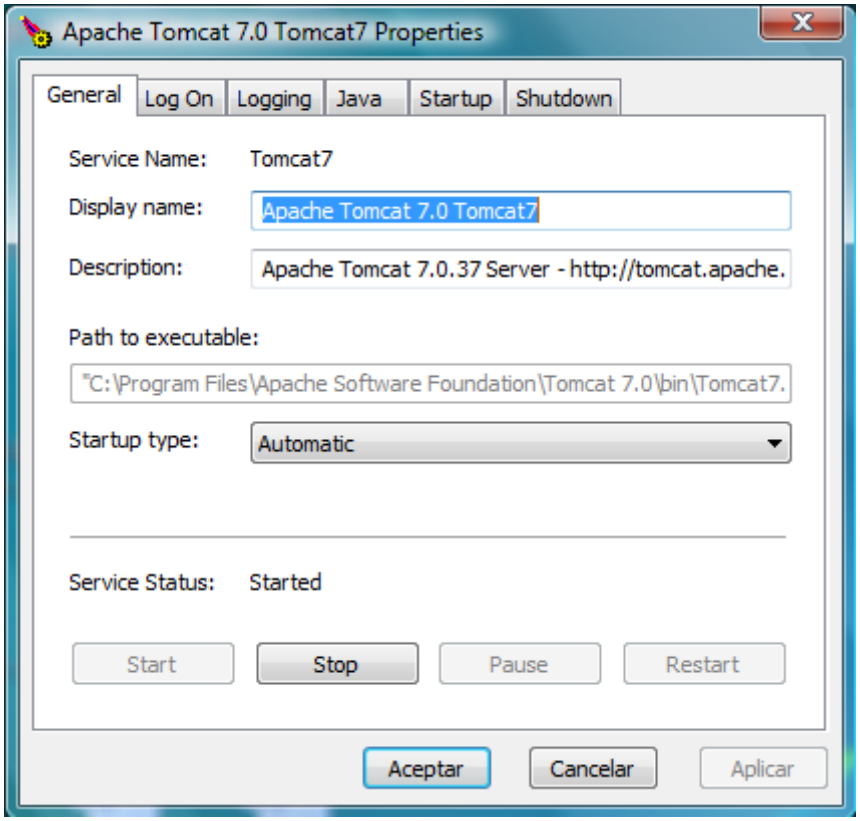
Manual Técnico

Índice

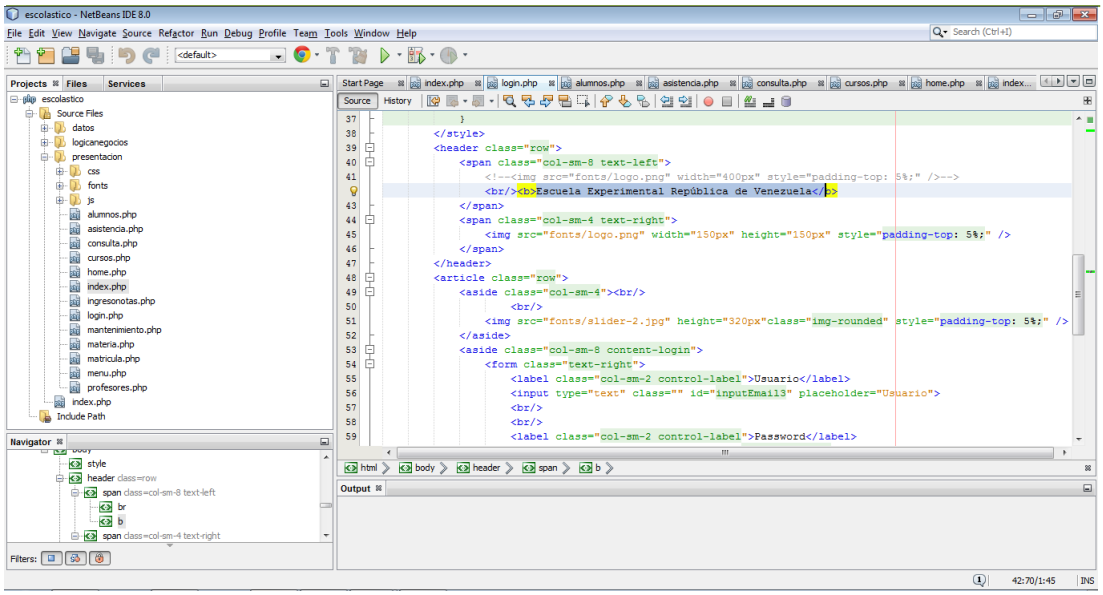
Titulo	Página
1. Apache Tomcat	102
2. Netbeans	102
3. SQL Yog	103
4. Rational Rose	104
5.- Scrip Base de Datos utilizada	104

Estos son los programas utilizados para el desarrollo de la aplicación:

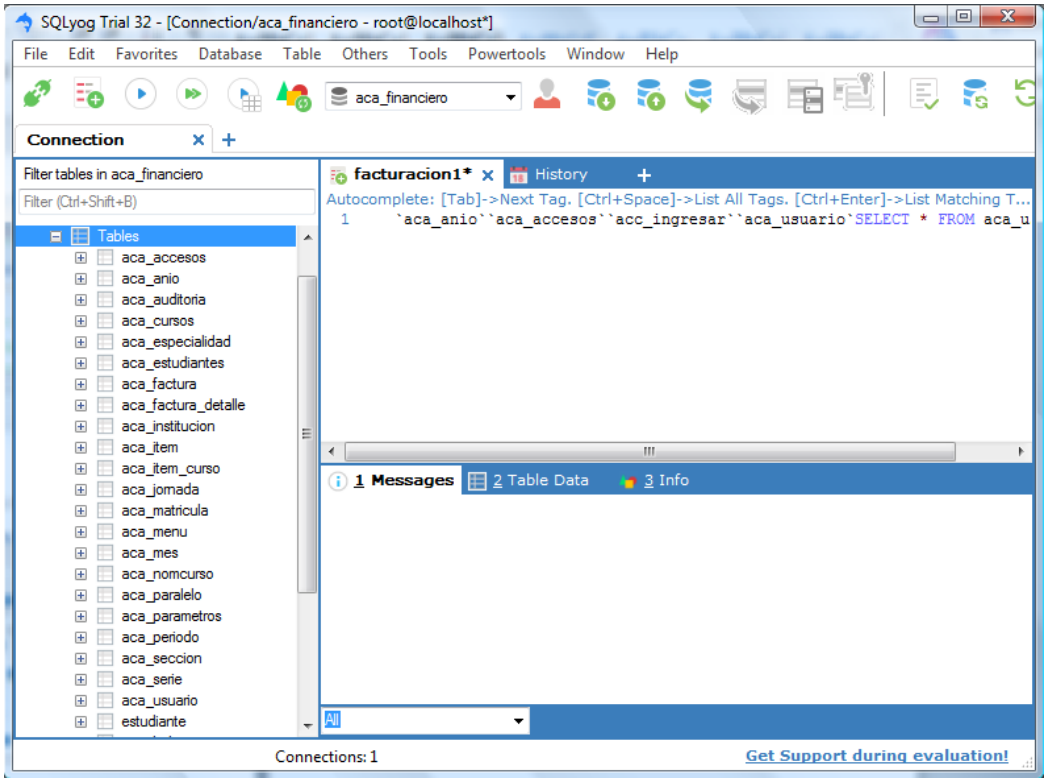
Apache Tomcat.- Es un servidor independiente de plataforma de código libre. Te permite instalar de forma sencilla Apache en tu propio ordenador, sin importar tu sistema operativo (Linux, Windows, MAC o Solaris). Y lo mejor de todo es que su uso es gratuito.



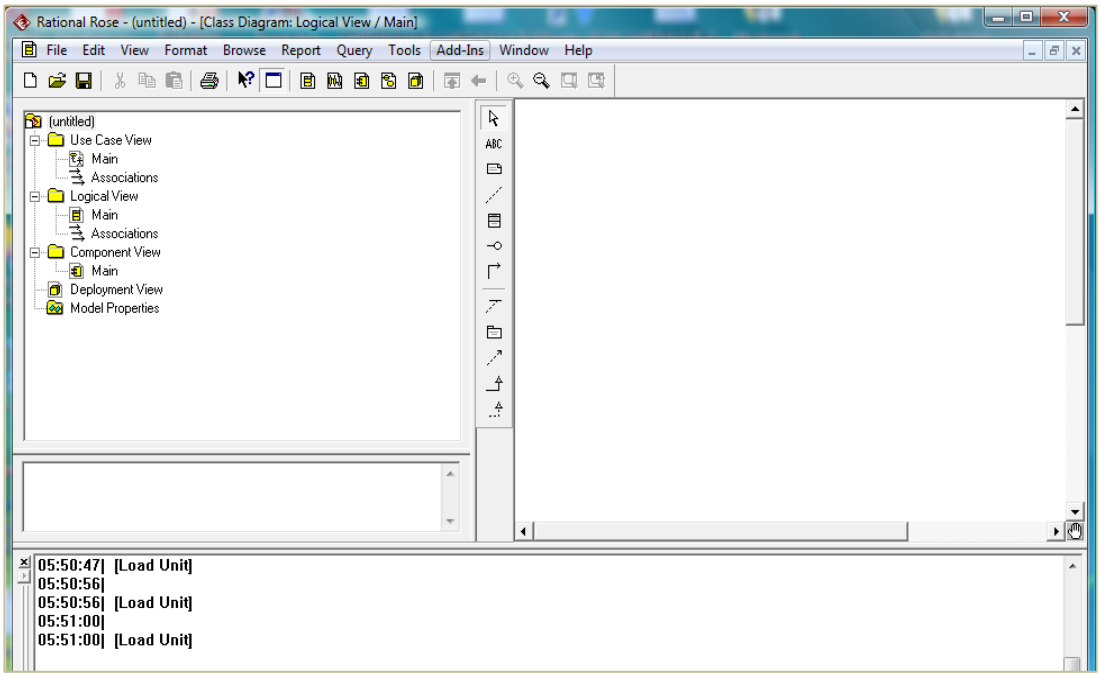
Netbeans.- Plataforma multilenguaje permite el desarrollo de Sistemas de escritorio como en ambiente web.



SQL Yog: Administrador de Base de datos en General, es decir multiplataforma.



RATIONAL ROSE.- Modelador para diseño de Software orientado a objetos que permite el desarrollo de modelos, casos y diagramas para la elaboración de cualquier sistema en UML.



SCRIP DE BASE DE DATOS UTILIZADA

/*

SQLyog Ultimate v8.61

MySQL - 5.5.35 : Database - aca_financiero

****

*****

***/**

/*!40101 SET NAMES utf8 */;

/*!40101 SET SQL_MODE=''*/;

```
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;

/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;

/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;

/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

CREATE DATABASE /*!32312 IF NOT EXISTS*/`aca_financiero` /*!40100 DEFAULT CHARACTER SET utf8 */;


USE `aca_financiero`;


/*Table structure for table `aca_accesos` */


DROP TABLE IF EXISTS `aca_accesos`;


CREATE TABLE `aca_accesos` (
  `acc_codigo` int(11) NOT NULL,
  `acc_modulo` varchar(3) DEFAULT NULL,
  `acc_ingresar` tinyint(1) DEFAULT NULL,
  `acc_agregar` tinyint(1) DEFAULT NULL,
  `acc_modificar` tinyint(1) DEFAULT NULL,
  `acc_eliminar` tinyint(1) DEFAULT NULL,
  `acc_perfil` varchar(3) DEFAULT NULL,
  `menu_codigo` int(11) DEFAULT NULL,
  PRIMARY KEY (`acc_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
/*Data for the table `aca_accesos` */

/*Table structure for table `aca_anio` */

DROP TABLE IF EXISTS `aca_anio`;

CREATE TABLE `aca_anio` (
  `ani_codigo` int(11) NOT NULL,
  `ani_nombre` varchar(5) DEFAULT NULL,
  PRIMARY KEY (`ani_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_anio` */

insert into `aca_anio`(`ani_codigo`,`ani_nombre`) values (1,'2014'),(2,'2015'),(3,'2016');

CREATE TABLE `aca_cursos` (
  `cur_codigo` int(11) NOT NULL,
  `per_codigo` int(11) DEFAULT NULL,
  `ncur_codigo` int(11) DEFAULT NULL,
  `par_codigo` int(11) DEFAULT NULL,
  `esp_codigo` int(11) DEFAULT NULL,
  `cur_fechai` date DEFAULT NULL,
  `cur_fechaf` date DEFAULT NULL,
  PRIMARY KEY (`cur_codigo`),
  KEY `FK_aca_periodo` (`per_codigo`),
```

```
  KEY `FK_aca_ncurso` (`ncur_codigo`),
  KEY `FK_aca_paralelo` (`par_codigo`),
  KEY `FK_aca_especialidad` (`esp_codigo`),
  CONSTRAINT `FK_aca_especialidad` FOREIGN KEY (`esp_codigo`) REFERENCES `aca_especialidad` (`esp_codigo`),
  CONSTRAINT `FK_aca_ncurso` FOREIGN KEY (`ncur_codigo`) REFERENCES `aca_nomcurso` (`ncur_codigo`),
  CONSTRAINT `FK_aca_paralelo` FOREIGN KEY (`par_codigo`) REFERENCES `aca_paralelo` (`par_codigo`),
  CONSTRAINT `FK_aca_periodo` FOREIGN KEY (`per_codigo`) REFERENCES `aca_periodo` (`per_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_cursos` */

/*Table structure for table `aca_factura` */

DROP TABLE IF EXISTS `aca_factura`;

CREATE TABLE `aca_factura` (
  `fac_codigo` int(11) NOT NULL,
  `fac_tipo_documento` varchar(5) DEFAULT NULL,
  `fac_numero` varchar(20) DEFAULT NULL,
  `fac_emision` date DEFAULT NULL,
  `fac_anulado` tinyint(1) DEFAULT NULL,
  `mat_codigo` int(11) DEFAULT NULL,
  `fac_fecha_anula` date DEFAULT NULL,
  `fac_subtotal` decimal(10,2) DEFAULT NULL,
```

```
  `fac_por_descuento` decimal(5,2) DEFAULT NULL,
  `fac_valor_descuento` decimal(10,2) DEFAULT NULL,
  `fac_valor_con_iva` decimal(10,2) DEFAULT NULL,
  `fac_valor_sin_iva` decimal(10,2) DEFAULT NULL,
  `fac_por_iva` decimal(8,2) DEFAULT NULL,
  `fac_valor_iva` decimal(10,2) DEFAULT NULL,
  `fac_valor_total` decimal(10,2) DEFAULT NULL,
  `fac_autoriza_sri` tinyint(1) DEFAULT NULL,
  `fac_fecha_sri` datetime DEFAULT NULL,
  `fac_autorizacion` varchar(100) DEFAULT NULL,
  `fac_clave_acceso` varchar(200) DEFAULT NULL,
  `fac_ruc` varchar(13) DEFAULT NULL,
  `fac_nombre` varchar(100) DEFAULT NULL,
  `fac_direccion` varchar(200) DEFAULT NULL,
  `fac_telefono` varchar(20) DEFAULT NULL,
  `fac_email` varchar(200) DEFAULT NULL,
  `fac_fecha_carga` datetime DEFAULT NULL,
  `fac_nombre_banco` varchar(50) DEFAULT NULL,
  `fac_codigo_banco` varchar(10) DEFAULT NULL,
  `fac_nombre_archivo` varchar(500) DEFAULT NULL,
  PRIMARY KEY (`fac_codigo`),
  KEY `FK_aca_factura_estudiante_matricula` (`mat_codigo`),
  CONSTRAINT `FK_aca_factura_estudiante_matricula` FOREIGN KEY (`mat_codigo`) REFERENCES `aca_matricula` (`mat_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/*Data for the table `aca_factura` */
```

```
/*Table structure for table `aca_factura_detalle` */

DROP TABLE IF EXISTS `aca_factura_detalle`;

CREATE TABLE `aca_factura_detalle` (
  `fac_det_codigo` int(11) NOT NULL,
  `fac_codigo` int(11) DEFAULT NULL,
  `item_codigo` int(11) NOT NULL,
  `mes_codigo` int(11) DEFAULT NULL,
  `fac_det_cantidad` decimal(10,4) DEFAULT NULL,
  `fac_det_por_descuento` decimal(5,2) DEFAULT NULL,
  `fac_det_valor_descuento` decimal(10,4) DEFAULT NULL,
  `fac_det_valor_unitario` decimal(10,4) DEFAULT NULL,
  `fac_det_por_iva` decimal(5,2) DEFAULT NULL,
  `fac_det_valor_iva` decimal(10,4) DEFAULT NULL,
  `fac_det_valor_total` decimal(10,4) DEFAULT NULL,
  PRIMARY KEY (`fac_det_codigo`),
  KEY `FK_aca_factura_detalle_codigo` (`fac_codigo`),
  KEY `FK_aca_factura_detalle_MES` (`mes_codigo`),
  KEY `FK_aca_factura_detalle_item` (`item_codigo`),
  CONSTRAINT `FK_aca_factura_detalle_codigo` FOREIGN KEY (`fac_codigo`) REFERENCES `aca_factura` (`fac_codigo`),
  CONSTRAINT `FK_aca_factura_detalle_item` FOREIGN KEY (`item_codigo`) REFERENCES `aca_item` (`item_codigo`),
  CONSTRAINT `FK_aca_factura_detalle_MES` FOREIGN KEY (`mes_codigo`) REFERENCES `aca_mes` (`mes_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

/*Data for the table `aca_factura_detalle` */

/*Table structure for table `aca_institucion` */

```
DROP TABLE IF EXISTS `aca_institucion`;

CREATE TABLE `aca_institucion` (
  `ins_codigo` int(11) NOT NULL,
  `ins_siglas` varchar(50) DEFAULT NULL,
  `ins_nombre` varchar(200) DEFAULT NULL,
  `ins_nombre2` varchar(100) DEFAULT NULL,
  `ins_direccion` varchar(100) DEFAULT NULL,
  `ins_lugar` varchar(100) DEFAULT NULL,
  `ins_telefono1` varchar(13) DEFAULT NULL,
  `ins_telefono2` varchar(13) DEFAULT NULL,
  `ins_fax` varchar(13) DEFAULT NULL,
  `ins_email` varchar(100) DEFAULT NULL,
  `ins_rector` varchar(100) DEFAULT NULL,
  `ins_secretaria` varchar(100) DEFAULT NULL,
  `ins_inspector` varchar(100) DEFAULT NULL,
  `ins_sello` longblob,
  `ins_fondo` longblob,
  `ins_fondoCarne` longblob,
  `ins_clave` varchar(100) DEFAULT NULL,
  `ins_smtp` varchar(200) DEFAULT 'smtp.gmail.com',
  `ins_puerto` varchar(100) DEFAULT '587',
```

```
  PRIMARY KEY (`ins_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Table structure for table `aca_item` */

DROP TABLE IF EXISTS `aca_item`;

CREATE TABLE `aca_item` (
  `item_codigo` int(11) NOT NULL,
  `item_descripcion` varchar(100) DEFAULT NULL,
  `item_precioventa1` double(16,5) DEFAULT '0.00000',
  `item_iva` tinyint(1) DEFAULT '0',
  `item_fechacreacion` date DEFAULT '1990-01-01',
  `ins_codigo` int(11) DEFAULT NULL,
  PRIMARY KEY (`item_codigo`),
  KEY `FK_aca_producto_institucion` (`ins_codigo`),
  CONSTRAINT `FK_aca_producto_institucion` FOREIGN KEY (`ins_codigo`) REFERENCES `aca_institucion` (`ins_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/*Data for the table `aca_item` */

/*Table structure for table `aca_item_curso` */

DROP TABLE IF EXISTS `aca_item_curso`;

CREATE TABLE `aca_item_curso` (
```

```
`item_cur_codigo` int(11) NOT NULL,
`cur_codigo` int(11) DEFAULT NULL,
`mes_codigo` int(11) DEFAULT NULL,
`item_codigo` int(11) NOT NULL,
`pro_cur_valor` decimal(10,2) DEFAULT NULL,
`pro_cur_vence` date DEFAULT NULL,
PRIMARY KEY (`item_cur_codigo`),
KEY `FK_aca_producto_curso` (`cur_codigo`),
KEY `FK_aca_producto_curso_mes` (`mes_codigo`),
KEY `FK_aca_item_curso_item_producto` (`item_codigo`),
CONSTRAINT `FK_aca_item_curso_item_producto` FOREIGN KEY (`item_codigo`) REFERENCES `aca_item` (`item_codigo`),
CONSTRAINT `FK_aca_producto_curso` FOREIGN KEY (`cur_codigo`) REFERENCES `aca_cursos` (`cur_codigo`),
CONSTRAINT `FK_aca_producto_curso_mes` FOREIGN KEY (`mes_codigo`) REFERENCES `aca_mes` (`mes_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/*Data for the table `aca_item_curso` */

/*Table structure for table `aca_jornada` */

DROP TABLE IF EXISTS `aca_jornada`;

CREATE TABLE `aca_jornada` (
`jor_codigo` int(11) NOT NULL,
`jor_nombre` varchar(50) DEFAULT NULL,
PRIMARY KEY (`jor_codigo`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_jornada` */

insert into `aca_jornada`(`jor_codigo`,`jor_nombre`) values (0,'.'),(1,'Matutina'),(3,'Nocturna');

/*Table structure for table `aca_matricula` */

DROP TABLE IF EXISTS `aca_matricula`;

CREATE TABLE `aca_matricula` (
  `mat_codigo` int(11) NOT NULL,
  `est_codigo` int(11) DEFAULT NULL,
  `cur_codigo` int(11) DEFAULT NULL,
  `mat_matricula` int(11) DEFAULT NULL,
  `mat_fechaMat` date DEFAULT NULL,
  `mat_fechaIns` date DEFAULT NULL,
  `mat_fechaRet` date DEFAULT NULL,
  `mat_fechaCam` date DEFAULT NULL,
  `mat_folio` int(11) DEFAULT NULL,
  `mat_estado` varchar(1) DEFAULT NULL,
  `mat_numeroMat` int(11) DEFAULT NULL,
  `mat_observacion1` varchar(100) DEFAULT NULL,
  `mat_observacion2` varchar(100) DEFAULT NULL,
  `mat_aprobado` tinyint(1) DEFAULT NULL,
  `mat_estadoMat` varchar(1) DEFAULT NULL,
```

```
 `mat_secretaria` tinyint(1) DEFAULT '0',
 `mat_colecturia` tinyint(1) DEFAULT '0',
 `mat_guia` varchar(7) DEFAULT NULL,
 `mat_fecha` date DEFAULT NULL,
 `mat_nueva` tinyint(1) DEFAULT '0',
 `mat_valor` double(8,4) DEFAULT '0.0000',
 `mat_porcentaje` double(8,2) DEFAULT '0.00',
 `mat_descuento` double(8,2) DEFAULT '0.00',
 `mat_recorrido` tinyint(1) DEFAULT '0',
 `mat_bus` varchar(100) DEFAULT NULL,
 `mat_matriculadopor` varchar(200) DEFAULT ' ',
 `mat_alimentacion` tinyint(1) DEFAULT '1',
 PRIMARY KEY (`mat_codigo`),
 KEY `FK_aca_curso` (`cur_codigo`),
 KEY `FK_aca_estudiante` (`est_codigo`),
 CONSTRAINT `FK_aca_curso` FOREIGN KEY (`cur_codigo`) REFERENCES `aca_cursos` (`cur_codigo`),
 CONSTRAINT `FK_aca_estudiante` FOREIGN KEY (`est_codigo`) REFERENCES `aca_estudiantes` (`est_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_matricula` */

/*Table structure for table `aca_menu` */

DROP TABLE IF EXISTS `aca_menu`;

CREATE TABLE `aca_menu` (
```

```
  `menu_codigo` int(11) NOT NULL,
  `menu_nombre` varchar(50) DEFAULT NULL,
  `menu_nivel` varchar(10) DEFAULT NULL,
  PRIMARY KEY (`menu_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/*Data for the table `aca_menu` */

/*Table structure for table `aca_mes` */

DROP TABLE IF EXISTS `aca_mes`;

CREATE TABLE `aca_mes` (
  `mes_codigo` int(11) NOT NULL,
  `mes_nombre` varchar(30) DEFAULT NULL,
  `mes_orden` int(11) DEFAULT NULL,
  `mes_estado` tinyint(1) DEFAULT '0',
  PRIMARY KEY (`mes_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_mes` */

/*Table structure for table `aca_nomcurso` */

DROP TABLE IF EXISTS `aca_nomcurso`;

CREATE TABLE `aca_nomcurso` (
```

```
  `ncur_codigo` int(11) NOT NULL,
  `ncur_nombre` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`ncur_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_nomcurso` */

/*Table structure for table `aca_paralelo` */

DROP TABLE IF EXISTS `aca_paralelo`;

CREATE TABLE `aca_paralelo` (
  `par_codigo` int(11) NOT NULL,
  `par_nombre` varchar(30) DEFAULT NULL,
  PRIMARY KEY (`par_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_paralelo` */

insert into `aca_paralelo`(`par_codigo`,`par_nombre`) values
(1,'\"A\"'),(2,'\"B\"'),(3,'\"C\"'),(4,'\"D\"'),(5,'\"E\"'),(6,'\"F\"');

/*Table structure for table `aca_parametros` */

DROP TABLE IF EXISTS `aca_parametros`;

CREATE TABLE `aca_parametros` (
```

```
  `par_secuencial` int(11) NOT NULL,
  `per_codigo` int(11) DEFAULT NULL,
  `par_variable` varchar(50) DEFAULT NULL,
  `par_descripcion` varchar(100) DEFAULT NULL,
  `par_tipo` varchar(1) DEFAULT NULL,
  `par_cvalor` varchar(100) DEFAULT NULL,
  `par_nvalor` double(12,2) DEFAULT NULL,
  `par_lvalor` tinyint(1) DEFAULT NULL,
  `par_dvalor` date DEFAULT NULL,
  PRIMARY KEY (`par_secuencial`),
  KEY `FK_aca_periododfe` (`per_codigo`),
  CONSTRAINT `FK_aca_periododfe` FOREIGN KEY (`per_codigo`) REFERENCES `aca_periodo` (`per_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_parametros` */

/*Table structure for table `aca_periodo` */

DROP TABLE IF EXISTS `aca_periodo`;

CREATE TABLE `aca_periodo` (
  `per_codigo` int(11) NOT NULL,
  `ins_codigo` int(11) DEFAULT NULL,
  `ani_codigo` int(11) DEFAULT NULL,
  `sec_codigo` int(11) DEFAULT NULL,
```

```
  `jor_codigo` int(11) DEFAULT NULL,
  `per_estado` tinyint(1) DEFAULT NULL,
  PRIMARY KEY (`per_codigo`),
  KEY `FK_aca_institucion` (`ins_codigo`),
  KEY `FK_aca_anio` (`ani_codigo`),
  KEY `FK_aca_seccion` (`sec_codigo`),
  KEY `FK_aca_jornada` (`jor_codigo`),
  CONSTRAINT `FK_aca_anio` FOREIGN KEY (`ani_codigo`) REFERENCES `aca_anio` (`ani_codigo`),
  CONSTRAINT `FK_aca_institucion` FOREIGN KEY (`ins_codigo`) REFERENCES `aca_institucion` (`ins_codigo`),
  CONSTRAINT `FK_aca_jornada` FOREIGN KEY (`jor_codigo`) REFERENCES `aca_jornada` (`jor_codigo`),
  CONSTRAINT `FK_aca_seccion` FOREIGN KEY (`sec_codigo`) REFERENCES `aca_seccion` (`sec_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_periodo` */

/*Table structure for table `aca_seccion` */

DROP TABLE IF EXISTS `aca_seccion`;

CREATE TABLE `aca_seccion` (
  `sec_codigo` int(11) NOT NULL,
  `sec_nombre` varchar(30) DEFAULT NULL,
  PRIMARY KEY (`sec_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
/*Data for the table `aca_seccion` */

insert into `aca_seccion`(`sec_codigo`,`sec_nombre`) values (1,'UNICA'),(2,'SECUNDARIA');

/*Table structure for table `aca_serie` */

DROP TABLE IF EXISTS `aca_serie`;

CREATE TABLE `aca_serie` (
  `serie_codigo` int(11) NOT NULL,
  `serie_tipo` varchar(10) DEFAULT NULL,
  `serie_autorizacion` varchar(20) DEFAULT NULL,
  `serie_desde` bigint(20) DEFAULT NULL,
  `serie_hasta` bigint(20) DEFAULT NULL,
  `serie_emision` datetime DEFAULT NULL,
  `serie_vence` datetime DEFAULT NULL,
  `serie_actual` bigint(20) DEFAULT NULL,
  `serie_longitud` int(11) DEFAULT NULL,
  `serie_activo` tinyint(1) DEFAULT NULL,
  `serie_inicial` varchar(20) DEFAULT NULL,
  `ins_codigo` int(11) DEFAULT NULL,
  PRIMARY KEY (`serie_codigo`),
  KEY `FK_aca_serie_institucion` (`ins_codigo`),
  CONSTRAINT `FK_aca_serie_institucion` FOREIGN KEY (`ins_codigo`) REFERENCES `aca_institucion` (`ins_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
/*Data for the table `aca_serie` */

/*Table structure for table `aca_usuario` */

DROP TABLE IF EXISTS `aca_usuario`;

CREATE TABLE `aca_usuario` (
  `usu_codigo` int(11) NOT NULL,
  `usu_nombre` varchar(50) DEFAULT NULL,
  `usu_apellido` varchar(50) DEFAULT NULL,
  `usu_direccion` varchar(60) DEFAULT NULL,
  `usu_telefono1` varchar(13) DEFAULT NULL,
  `usu_telefono2` varchar(13) DEFAULT NULL,
  `usu_tipoIdentificacion` varchar(1) DEFAULT NULL,
  `usu_genero` varchar(1) DEFAULT NULL,
  `usu_fechaIngreso` date DEFAULT NULL,
  `usu_estado` tinyint(1) DEFAULT NULL,
  `usu_fechaRetiro` date DEFAULT NULL,
  `usu_email` varchar(100) DEFAULT NULL,
  `usu_usuario` varchar(30) DEFAULT NULL,
  `usu_clave` varchar(200) DEFAULT NULL,
  `usu_tipoUsuario` int(11) DEFAULT NULL,
  `usu_tratamiento` varchar(5) DEFAULT NULL,
  `usu_perfil` varchar(3) DEFAULT NULL,
  `usu_identificacion` varchar(13) DEFAULT NULL,
  `usu_pregunta` varchar(100) DEFAULT NULL,
```

```
  `usu_respuesta` varchar(100) DEFAULT NULL,
  `usu_cambioClave` tinyint(1) DEFAULT '0',
  `usu_civil` varchar(2) DEFAULT 'S',
  `usu_sueldo` double(8,2) DEFAULT '0.00',
  PRIMARY KEY (`usu_codigo`),
  UNIQUE KEY `idx_cedula` (`usu_identificacion`),
  UNIQUE KEY `INDEX_USUARIO_USERID` (`usu_usuario`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `aca_usuario` */

insert  into
/*Table structure for table `estudiante` */

DROP TABLE IF EXISTS `estudiante`;

CREATE TABLE `estudiante` (
  `CODIGO` double DEFAULT NULL,
  `NOMBRE` varchar(255) DEFAULT NULL,
  `APELLIDOS` varchar(255) DEFAULT NULL,
  `dir_repres` varchar(255) DEFAULT NULL,
  `tel_repres` varchar(255) DEFAULT NULL,
  `alumno` varchar(255) DEFAULT NULL,
  `tipoiFAC` varchar(255) DEFAULT NULL,
  `ruc_repres` varchar(255) DEFAULT NULL,
  `nom_repres` varchar(255) DEFAULT NULL,
  `dir_repres1` varchar(255) DEFAULT NULL,
```

```
  `tel_repres1` varchar(255) DEFAULT NULL,
  `EMAIL FAC` varchar(255) DEFAULT NULL,
  `nom_grado` varchar(255) DEFAULT NULL,
  `nombre1` varchar(255) DEFAULT NULL,
  `nuevo` bit(1) NOT NULL,
  `nue_grado` varchar(255) DEFAULT NULL,
  `cod_curso` varchar(255) DEFAULT NULL,
  `cod_cu_otr` double DEFAULT NULL,
  `sexo` varchar(255) DEFAULT NULL,
  `registrado` bit(1) NOT NULL,
  `tip_sri` double DEFAULT NULL,
  `cur_migra` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/*Data for the table `estudiante` */


/*Table structure for table `mysql_data` */

DROP TABLE IF EXISTS `mysql_data`;

CREATE TABLE `mysql_data` (
  `conf_codigo` int(11) NOT NULL,
  `conf_empresas` varchar(100) DEFAULT NULL,
  `conf_finicio` varchar(100) DEFAULT NULL,
  `conf_ffin` varchar(100) DEFAULT NULL,
  `conf_status` varchar(100) DEFAULT NULL,
```

```
  PRIMARY KEY (`conf_codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `mysql_data` */

/*Table structure for table `tmp_tempral` */

DROP TABLE IF EXISTS `tmp_tempral`;

CREATE TABLE `tmp_tempral` (
  `est_codigo` int(11) DEFAULT NULL,
  `mat_codigo` int(11) NOT NULL,
  `fac_codigo` int(11) NOT NULL,
  `cur_codigo` int(11) NOT NULL,
  `per_codigo` int(11) NOT NULL,
  `item_codigo` int(11) NOT NULL,
  `mes_codigo` int(11) DEFAULT NULL,
  `fac_det_codigo` int(11) NOT NULL,
  `ncur_nombre` varchar(100) CHARACTER SET latin1 DEFAULT NULL,
  `par_nombre` varchar(30) CHARACTER SET latin1 DEFAULT NULL,
  `ani_nombre` varchar(5) CHARACTER SET latin1 DEFAULT NULL,
  `fac_det_cantidad` decimal(10,4) DEFAULT NULL,
  `fac_det_valor_unitario` decimal(10,4) DEFAULT NULL,
  `fac_det_por_iva` decimal(5,2) DEFAULT NULL,
  `fac_det_valor_iva` decimal(10,4) DEFAULT NULL,
  `fac_det_valor_total` decimal(10,4) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

/*Data for the table `tmp_tempral` */

/* Procedure structure for procedure `spEstadoCuenta` */

/*!50003 DROP PROCEDURE IF EXISTS `spEstadoCuenta` */;

DELIMITER $$

/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `spEstadoCuenta`(IN estudianteCodigo INT)

BEGIN

DROP TEMPORARY TABLE IF EXISTS tmp_estado_cuenta;

CREATE TEMPORARY TABLE tmp_estado_cuenta AS

SELECT
mat.est_codigo,mat.mat_codigo,cfac.fac_codigo,cur.cur_codigo,per.per_codigo,
cfdet.item_codigo,cfdet.mes_codigo,cfdet.fac_det_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,cfdet.fac_det_cantidad,cfdet.fac_det_valor_unitario,

cfdet.fac_det_por_iva,cfdet.fac_det_valor_iva,cfdet.fac_det_valor_total,item.item_descripcion,mes.mes_nombre,

cFac.fac_numero

FROM aca_factura cFac,aca_factura_detalle cFdet,aca_matricula mat,aca_estudiantes est,aca_cursos cur,aca_periodo per,

aca_nomcurso curn,aca_paralelo par,aca_anio anio,aca_item item,aca_mes mes

WHERE cFac.fac_codigo = cFdet.fac_codigo

AND cFac.mat_codigo = mat.mat_codigo

AND mat.est_codigo = est.est_codigo

AND mat.cur_codigo = cur.cur_codigo

```
AND cur.per_codigo = per.per_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND cur.par_codigo = par.par_codigo

AND per.ani_codigo = anio.ani_codigo

AND cfdet.item_codigo = item.item_codigo

AND cfdet.mes_codigo = mes.mes_codigo

AND mat.est_codigo = estudianteCodigo

;

SELECT mat.est_codigo,mat.mat_codigo,
estC.fac_numero,cur.cur_codigo,per.per_codigo,itemC.item_codigo,itemC.mes_
codigo,

estC.fac_det_codigo,curn.ncur_nombre,par.par_nombre,anio.ani_nombre,

estC.fac_det_cantidad,itemC.pro_cur_valor AS fac_det_valor_unitario,

estC.fac_det_valor_iva,-- itemC.pro_cur_valor AS fac_det_valor_total,

(itemC.pro_cur_valor - IFNULL(estC.fac_det_valor_total,0))AS
fac_det_valor_total,

item.item_descripcion,mes.mes_nombre

FROM aca_item item,aca_matricula mat,aca_estudiantes est,aca_anio
anio,aca_cursos cur,aca_nomcurso curn,aca_paralelo par,aca_periodo
per,aca_mes mes,aca_item_curso itemC

LEFT JOIN tmp_estado_cuenta estC ON itemC.cur_codigo = estC.cur_codigo

AND itemC.mes_codigo = estC.mes_codigo AND itemC.item_codigo =
estC.item_codigo

WHERE item.item_codigo = itemC.item_codigo

AND itemC.cur_codigo = mat.cur_codigo

AND mat.est_codigo = est.est_codigo

AND per.per_codigo = cur.per_codigo

AND cur.cur_codigo = mat.cur_codigo

AND per.ani_codigo = anio.ani_codigo
```

```
AND par.par_codigo = cur.par_codigo

AND mes.mes_codigo = itemC.mes_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND mat.est_codigo = estudianteCodigo ;

END */$$

DELIMITER ;


/* Procedure structure for procedure `spEstadoCuentaBanco` */


/*!50003 DROP PROCEDURE IF EXISTS  `spEstadoCuentaBanco` */;


DELIMITER $$


/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `spEstadoCuentaBanco`(IN mesCodigo INT)

BEGIN

DECLARE mesOrdenado INT DEFAULT 0 ;

SELECT mes_orden INTO mesOrdenado FROM aca_mes WHERE mes_codigo = mesCodigo;

DROP TEMPORARY TABLE IF EXISTS tmp_estado_cuenta_B;

CREATE TEMPORARY TABLE  tmp_estado_cuenta_B AS

SELECT
mat.est_codigo,mat.mat_codigo,cur.cur_codigo,per.per_codigo,cfdet.mes_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,mes.mes_nombre,

SUM(cFac.fac_valor_total) AS facturado

FROM aca_factura cFac,aca_factura_detalle cFdet,aca_matricula mat,aca_estudiantes est,aca_cursos cur,aca_periodo per,

aca_nomcurso curn,aca_paralelo par,aca_anio anio,aca_item item,aca_mes mes
```

```
WHERE cFac.fac_codigo = cFdet.fac_codigo

AND cFac.mat_codigo = mat.mat_codigo

AND mat.est_codigo = est.est_codigo

AND mat.cur_codigo = cur.cur_codigo

AND cur.per_codigo = per.per_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND cur.par_codigo = par.par_codigo

AND per.ani_codigo = anio.ani_codigo

AND cfdet.item_codigo = item.item_codigo

AND cfdet.mes_codigo = mes.mes_codigo

AND mes.mes_orden <= mesOrdenado

;

DROP TEMPORARY TABLE IF EXISTS tmp_estado_cuenta_SINBECA;

CREATE TEMPORARY TABLE  tmp_estado_cuenta_SINBECA AS

SELECT
mat.mat_matricula,mat.est_codigo,est.est_nombre,est.est_apellido,mat.mat_cod
igo,cur.cur_codigo,per.per_codigo,itemC.mes_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,mes.mes_nombre,SUM(it
emC.pro_cur_valor - IFNULL(estC.facturado,0)) AS fac_det_valor_total

FROM aca_item item,aca_matricula mat,aca_estudiantes est,aca_anio
anio,aca_cursos cur,aca_nomcurso curn,

aca_paralelo par,aca_periodo per,aca_mes mes,aca_item_curso itemC

LEFT JOIN tmp_estado_cuenta_B estC ON itemC.cur_codigo =
estC.cur_codigo

AND itemC.mes_codigo = estC.mes_codigo

WHERE item.item_codigo = itemC.item_codigo

AND itemC.cur_codigo = mat.cur_codigo

AND mat.est_codigo = est.est_codigo

AND per.per_codigo = cur.per_codigo
```

```
AND cur.cur_codigo = mat.cur_codigo

AND per.ani_codigo = anio.ani_codigo

AND par.par_codigo = cur.par_codigo

AND mes.mes_codigo = itemC.mes_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND mes.mes_orden <= mesOrdenado

-- AND estC.fac_det_codigo IS NULL

GROUP BY
mat.mat_matricula,mat.est_codigo,est.est_nombre,est.est_apellido,mat.mat_cod
igo,cur.cur_codigo,per.per_codigo,itemC.mes_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,mes.mes_nombre

HAVING SUM(itemC.pro_cur_valor - IFNULL(estC.facturado,0)) > 0;

SELECT
dt.mat_matricula,dt.est_codigo,dt.est_nombre,dt.est_apellido,dt.mat_codigo,dt.
cur_codigo,dt.per_codigo,dt.mes_codigo,

dt.ncur_nombre,dt.par_nombre,dt.ani_nombre,dt.mes_nombre,dt.fac_det_valo
r_total - mat.mat_valor AS fac_det_valor_total

FROM tmp_estado_cuenta_SINBECA dt, aca_matricula mat

WHERE mat.mat_codigo = dt.mat_codigo

and (dt.fac_det_valor_total - mat.mat_valor) > 0 ;

END */$$

DELIMITER ;


/* Procedure structure for procedure `spEstadoCuentaDetalle` */


/*!50003 DROP PROCEDURE IF EXISTS  `spEstadoCuentaDetalle` */;


DELIMITER $$
```

```
/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `spEstadoCuentaDetalle`(IN estudianteCodigo INT,in mesCodigo int)

BEGIN

DROP TEMPORARY TABLE IF EXISTS tmp_estado_cuenta_det;

CREATE TEMPORARY TABLE  tmp_estado_cuenta_det AS

SELECT
mat.est_codigo,mat.mat_codigo,cfac.fac_codigo,cur.cur_codigo,per.per_codigo,
cfdet.item_codigo,cfdet.mes_codigo,cfdet.fac_det_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,cfdet.fac_det_cantidad,cf
det.fac_det_valor_unitario,

cfdet.fac_det_por_iva,cfdet.fac_det_valor_iva,cfdet.fac_det_valor_total,item.ite
m_descripcion,mes.mes_nombre

FROM aca_factura cFac,aca_factura_detalle cFdet,aca_matricula
mat,aca_estudiantes est,aca_cursos cur,aca_periodo per,

aca_nomcurso curn,aca_paralelo par,aca_anio anio,aca_item item,aca_mes mes

WHERE cFac.fac_codigo = cFdet.fac_codigo

AND cFac.mat_codigo = mat.mat_codigo

AND mat.est_codigo = est.est_codigo

AND mat.cur_codigo = cur.cur_codigo

AND cur.per_codigo = per.per_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND cur.par_codigo = par.par_codigo

AND per.ani_codigo = anio.ani_codigo

AND cfdet.item_codigo = item.item_codigo

AND cfdet.mes_codigo = mes.mes_codigo

AND mat.est_codigo = estudianteCodigo

and mes.mes_codigo = mesCodigo ;

SELECT mat.est_codigo,mat.mat_codigo, 0 AS
fac_codigo,cur.cur_codigo,per.per_codigo,itemC.item_codigo,itemC.mes_codig
o,
```

```
estC.fac_det_codigo,curn.ncur_nombre,par.par_nombre,anio.ani_nombre,

estC.fac_det_cantidad,itemC.pro_cur_valor AS fac_det_valor_unitario,

estC.fac_det_valor_iva, itemC.pro_cur_valor AS
fac_det_valor_total,item.item_descripcion,mes.mes_nombre

FROM aca_item item,aca_matricula mat,aca_estudiantes est,aca_anio
anio,aca_cursos cur,aca_nomcurso curn,aca_paralelo par,aca_periodo
per,aca_mes mes,aca_item_curso itemC

LEFT JOIN tmp_estado_cuenta_det estC ON itemC.cur_codigo =
estC.cur_codigo

AND itemC.mes_codigo = estC.mes_codigo AND itemC.item_codigo =
estC.item_codigo

WHERE item.item_codigo = itemC.item_codigo

AND itemC.cur_codigo = mat.cur_codigo

AND mat.est_codigo = est.est_codigo

AND per.per_codigo = cur.per_codigo

AND cur.cur_codigo = mat.cur_codigo

AND per.ani_codigo = anio.ani_codigo

AND par.par_codigo = cur.par_codigo

AND mes.mes_codigo = itemC.mes_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND mat.est_codigo = estudianteCodigo

and mes.mes_codigo = mesCodigo ;

END */$$

DELIMITER ;


/* Procedure structure for procedure `spEstadoCuentaMensual` */


/*!50003 DROP PROCEDURE IF EXISTS  `spEstadoCuentaMensual` */;
```

```
DELIMITER $$

/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `spEstadoCuentaMensual`(IN matMatricula INT,in mesCodigo int)

BEGIN

DROP TEMPORARY TABLE IF EXISTS tmp_estado_cuenta;

CREATE TEMPORARY TABLE  tmp_estado_cuenta AS

SELECT
mat.est_codigo,mat.mat_codigo,cfac.fac_codigo,cur.cur_codigo,per.per_codigo,
cfdet.item_codigo,cfdet.mes_codigo,cfdet.fac_det_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,cfdet.fac_det_cantidad,cf
det.fac_det_valor_unitario,

cfdet.fac_det_por_iva,cfdet.fac_det_valor_iva,cfdet.fac_det_valor_total,item.ite
m_descripcion,mes.mes_nombre

FROM aca_factura cFac,aca_factura_detalle cFdet,aca_matricula
mat,aca_estudiantes est,aca_cursos cur,aca_periodo per,

aca_nomcurso curn,aca_paralelo par,aca_anio anio,aca_item item,aca_mes mes

WHERE cFac.fac_codigo = cFdet.fac_codigo

AND cFac.mat_codigo = mat.mat_codigo

AND mat.est_codigo = est.est_codigo

AND mat.cur_codigo = cur.cur_codigo

AND cur.per_codigo = per.per_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND cur.par_codigo = par.par_codigo

AND per.ani_codigo = anio.ani_codigo

AND cfdet.item_codigo = item.item_codigo

AND cfdet.mes_codigo = mes.mes_codigo

AND mat.mat_matricula = matMatricula

and mes.mes_codigo = mesCodigo;
```

```
SELECT mat.est_codigo,mat.mat_codigo, 0 AS fac_codigo,cur.cur_codigo,per.per_codigo,itemC.item_codigo,itemC.mes_codigo,

estC.fac_det_codigo,curn.ncur_nombre,par.par_nombre,anio.ani_nombre,

estC.fac_det_cantidad,itemC.pro_cur_valor AS fac_det_valor_unitario,

estC.fac_det_valor_iva, itemC.pro_cur_valor AS fac_det_valor_total,item.item_descripcion,mes.mes_nombre

FROM aca_item item,aca_matricula mat,aca_estudiantes est,aca_anio anio,aca_cursos cur,aca_nomcurso curn,aca_paralelo par,aca_periodo per,aca_mes mes,aca_item_curso itemC

LEFT JOIN tmp_estado_cuenta estC ON itemC.cur_codigo = estC.cur_codigo

AND itemC.mes_codigo = estC.mes_codigo AND itemC.item_codigo = estC.item_codigo

WHERE item.item_codigo = itemC.item_codigo

AND itemC.cur_codigo = mat.cur_codigo

AND mat.est_codigo = est.est_codigo

AND per.per_codigo = cur.per_codigo

AND cur.cur_codigo = mat.cur_codigo

AND per.ani_codigo = anio.ani_codigo

AND par.par_codigo = cur.par_codigo

AND mes.mes_codigo = itemC.mes_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND mat.mat_matricula = matMatricula

and mes.mes_codigo = mesCodigo ;

END */$$

DELIMITER ;


/* Procedure structure for procedure `spEstadoCuentaRecibeBanco` */
```

```
/*!50003 DROP PROCEDURE IF EXISTS  `spEstadoCuentaRecibeBanco` */;

DELIMITER $$

/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `spEstadoCuentaRecibeBanco`(IN mesCodigo INT,IN estCodigo INT)

BEGIN

DROP TEMPORARY TABLE IF EXISTS tmp_estado_cuenta_B;

CREATE TEMPORARY TABLE  tmp_estado_cuenta_B AS

SELECT
mat.est_codigo,mat.mat_codigo,cfac.fac_codigo,cur.cur_codigo,per.per_codigo,
cfdet.item_codigo,cfdet.mes_codigo,cfdet.fac_det_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,cfdet.fac_det_cantidad,cf
det.fac_det_valor_unitario,

cfdet.fac_det_por_iva,cfdet.fac_det_valor_iva,cfdet.fac_det_valor_total,item.ite
m_descripcion,mes.mes_nombre

FROM aca_factura cFac,aca_factura_detalle cFdet,aca_matricula
mat,aca_estudiantes est,aca_cursos cur,aca_periodo per,

aca_nomcurso curn,aca_paralelo par,aca_anio anio,aca_item item,aca_mes mes

WHERE cFac.fac_codigo = cFdet.fac_codigo

AND cFac.mat_codigo = mat.mat_codigo

AND mat.est_codigo = est.est_codigo

AND mat.cur_codigo = cur.cur_codigo

AND cur.per_codigo = per.per_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND cur.par_codigo = par.par_codigo

AND per.ani_codigo = anio.ani_codigo

AND cfdet.item_codigo = item.item_codigo

AND cfdet.mes_codigo = mes.mes_codigo
```

```
AND mes.mes_codigo = mesCodigo

AND mat.mat_matricula = estCodigo

;

SELECT
mat.mat_matricula,mat.est_codigo,est.est_nombre,est.est_apellido,mat.mat_codigo,cur.cur_codigo,per.per_codigo,itemC.mes_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,mes.mes_nombre,SUM(itemC.pro_cur_valor - estC.fac_det_valor_total) AS fac_det_valor_total ,SUM(estC.fac_det_valor_total) AS total2

FROM aca_item item,aca_matricula mat,aca_estudiantes est,aca_anio anio,aca_cursos cur,aca_nomcurso curn,

aca_paralelo par,aca_periodo per,aca_mes mes,aca_item_curso itemC

LEFT JOIN tmp_estado_cuenta_B estC ON itemC.cur_codigo = estC.cur_codigo

AND itemC.mes_codigo = estC.mes_codigo AND itemC.item_codigo = estC.item_codigo

WHERE item.item_codigo = itemC.item_codigo

AND itemC.cur_codigo = mat.cur_codigo

AND mat.est_codigo = est.est_codigo

AND per.per_codigo = cur.per_codigo

AND cur.cur_codigo = mat.cur_codigo

AND per.ani_codigo = anio.ani_codigo

AND par.par_codigo = cur.par_codigo

AND mes.mes_codigo = itemC.mes_codigo

AND curn.ncur_codigo = cur.ncur_codigo

AND mes.mes_codigo = mesCodigo

AND mat.mat_matricula = estCodigo

-- AND estC.fac_det_codigo IS NULL
```

```
GROUP BY mat.mat_matricula,mat.est_codigo,est.est_nombre,est.est_apellido,mat.mat_codigo,cur.cur_codigo,per.per_codigo,itemC.mes_codigo,

curn.ncur_nombre,par.par_nombre,anio.ani_nombre,mes.mes_nombre

HAVING (SUM(itemC.pro_cur_valor - estC.fac_det_valor_total) - SUM(estC.fac_det_valor_total)) >0 ;

END */$$

DELIMITER ;


/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;

/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;

/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;

/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
```

A.07 Herramientas

Todas las herramientas utilizadas para la elaboración del proyecto son:

- Laptop Toshiba Core 2 Duo, 4 RAM, la cual se utiliza para la realización de la aplicación.
- Yog SQL el cual es la interfaz gráfica del MySQL.
- NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java.
- Rational Rose es una herramienta de modelamiento visual para el diseño de sistemas orientado a objetos principalmente.
- Power Designer es una herramienta inteligente para la construcción de base de datos
- Apache Tomcat es un contenedor web basado en lenguaje Java.
- Microsoft Office paquete de programas informáticos como hoja de cálculo, procesador de texto, etc.
- Microsoft Project es una herramienta para desarrollo de planes, dar seguimiento a proyectos, etc.
- Microsoft Visio software para dibujo de varios tipos como: planos, diagramas, cuadros conceptuales y figuras.

ENLACES BIBLIOGRAFICOS

http://sistemas.mag.go.cr/SoporteTecnico/Est%C3%A1ndares%20de%20Sistemas.pdf

https://docs.google.com/document/d/1UjUZJpNO9rc4uN98JKteih4JpoeY3lamjyWwxDKqcIg/edit?pli=1

http://www.fing.edu.uy/tecnoinf/mvd/cursos/ingsoft/material/teorico/is06-DisenioIU.pdf

https://docs.google.com/document/d/1UjUZJpNO9rc4uN98JKteih4JpoeY3lamjyWwxDKqcIg/edit?pli=1

http://www.informatica-juridica.com/trabajos/Procedimiento_para_la_realizacion_de_pruebas_de_unidad.asp

http://ocw.unican.es/ensenanzas-tecnicas/ingenieria-del-software-ii/materiales/tema1-pruebasSistemasSoftware.pdf

http://www.it.uc3m.es/ttrd/material/05-pruebas-de-programas.pdf

https://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=14&cad=rja&uact=8&ved=0CHIQFjAN&url=https%3A%2F%2Fadministracionelectronica.gob.es%2Fpae_Home%2Fdms%2Fpae_Home%2Fdocumentos%2FDocumentacion%2FMetodologias-y-guias%2FMetricav3%2FMETRICA_V3_Implantacion_y_Aceptacion_del_Sistema.pdf&ei=7_D7U871CKTesATYrYGgCg&usg=AFQjCNELLsQOgg-xIOir-Iw22gpQmXzNFg&sig2=p7lk8sbkQT1mHJED65yl8Q&bvm=bv.73612305,d.cWc

http://riunet.upv.es/bitstream/handle/10251/11307/Trabajo%20Fin%20Master.pdf?sequence=1

http://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&cad=rja&uact=8&ved=0CBsQFjAA&url=http%3A%2F%2Fcibertec.googlecode.com%2Ffiles%2FPrue

bas%2520de%2520Software.pdf&ei=oPP7U7fMJ5LKsQTi0ID4CQ&usg=AFQjCNH9A8Jn9qZn

86RRUkziI69nwLE7uA&sig2=l0-y_mRax-er9vROZioHwA&bvm=bv.73612305,d.cWc

http://www.serikat.es/pdf/revista_14/revista_14_03.pdf

http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/viveros_s_ca/capitulo6.pdf

http://qatecnico.blogspot.com/2012/03/pruebas-de-rendimiento-tipos-y.html