



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA
LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN
CARLOS DE GUATEMALA.**

Gloria Ercira Colindres Solorzano

Asesorado por el Ing. Jorge Armin Mazariegos Rabanales

Guatemala, noviembre de 2009

UNIVERSIDAD SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



**PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA
LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN
CARLOS DE GUATEMALA.**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

GLORIA ERCIRA COLINDRES SOLORIZANO

ASESORADO POR EL ING. JORGE ARMIN MAZARIEGOS RABANALES
AL CONFERÍRSELE EL TÍTULO DE

INGENIERA EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2009

UNIVERSIDAD SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Maritza Guerrero de López
VOCAL III	Ing. Miguel Angel Davila Calderón
VOCAL IV	Br. Milton De León Bran
VOCAL V	Br. Isaac Sultan Mejia
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

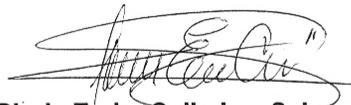
DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera
EXAMINADORA	Inga. Sonia Yolanda Castañeda Ramírez
EXAMINADOR	Ing. Marlon Antonio Pérez Turk
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración el trabajo de graduación titulado:

**PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA
LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN
CARLOS DE GUATEMALA,**

tema que me fuera asignado por la Coordinación de la Carrera de Ingeniería en Ciencias y Sistemas, en abril de 2007.



Gloria Ercira Colindres Solorzano

Guatemala, 14 de Noviembre de 2009

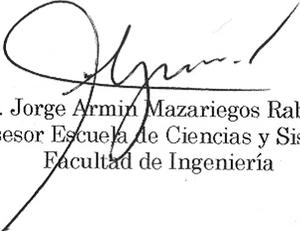
Inga. Norma Ileana Sarmiento Zeceña
Directora de Unidad de EPS
Facultad de Ingeniería
Universidad de San Carlos de Guatemala
Su despacho

Estimada Ingeniera Sarmiento:

Atentamente me dirijo a usted para informarle que fue Aprobado el contenido del documento final de EPS de la estudiante de la carrera de Ingeniería en Ciencias y Sistemas Gloria Ercira Colindres Solorzano con carné 9213167, en el proyecto: "PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADEMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA."

Sin otro particular, reciba un cordial saludo.

Atentamente,



Ing. Jorge Armin Mazariegos Rabanales
Asesor Escuela de Ciencias y Sistemas
Facultad de Ingeniería

C.C. Archivo

Universidad de San Carlos de Guatemala
Facultad de Ingeniería



UNIDAD DE E.P.S.

Guatemala, 03 de septiembre de 2009.
REF.EPS.DOC.1297.09.09.

Inga. Norma Ileana Sarmiento Zeceña de Serrano
Directora Unidad de EPS
Facultad de Ingeniería
Presente

Estimada Ingeniera Sarmiento Zeceña.

Por este medio atentamente le informo que como Supervisora de la Práctica del Ejercicio Profesional Supervisado, (E.P.S) de la estudiante universitaria de la Carrera de Ingeniería en Ciencias y Sistemas, **Gloria Ercira Colindres Solorzano** Carné No. **9213167** procedí a revisar el informe final, cuyo título es **“PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**.

En tal virtud, **LO DOY POR APROBADO**, solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

“Id y Enseñad a Todas”

Inga. Floriza Felipa Avila Pesquera de Medinilla
Supervisora de EPS
Área de Ingeniería en Ciencias y Sistemas

FFAPdM/RA



Edificio de E.P.S., Facultad de Ingeniería, Universidad de San Carlos de Guatemala
Ciudad Universitaria zona 12, teléfono directo: 2442-3509

Universidad de San Carlos de Guatemala
Facultad de Ingeniería



UNIDAD DE E.P.S.

Guatemala, 03 de septiembre de 2009.
REF.EPS.D.547.09.09.

Ing. Marlon Antonio Pérez Turck
Director Escuela de Ingeniería Ciencias y Sistemas
Facultad de Ingeniería
Presente

Estimado Ingeniero Perez Turck.

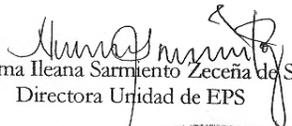
Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **“PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, que fue desarrollado por la estudiante universitaria **Gloria Ercira Colindres Solorzano** Carné No. **9213167** quien fue debidamente asesorada por el Ing. Jorge Armin Mazariegos y supervisada por la Inga. Floriza Felipa Ávila Pesquera de Medinilla

Por lo que habiendo cumplido con los objetivos y requisitos de ley del referido trabajo y existiendo la aprobación del mismo por parte del Asesor y de la Supervisora de EPS, en mi calidad de Directora apruebo su contenido solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

“Id y Enseñad a Todos”


Inga. Norma Ileana Sarmiento Zeceña de Serrano
Directora Unidad de EPS

NISZ/ra



Edificio de E.P.S., Facultad de Ingeniería, Universidad de San Carlos de Guatemala
Ciudad Universitaria zona 12, teléfono directo: 2442-3509



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 16 de Noviembre de 2009

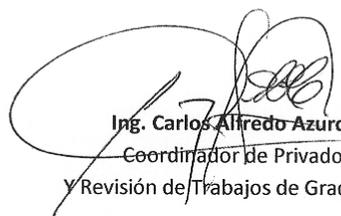
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación de la estudiante GLORIA ERCIRA COLINDRES SOLORZANO, titulado: **“PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado **“PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, presentado por la estudiante GLORIA ERCIRA COLINDRES SOLORZANO, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAR A TODOS”

Ing. *Martín Antonio Pérez Turiso*
Director, Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 16 de noviembre 2009

Universidad de San Carlos
de Guatemala



Facultad de Ingeniería
Decanato

Ref. DTG.397.09

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADÉMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por la estudiante universitaria **Gloria Ercira Colindres Solorzano**, autoriza la impresión del mismo.

IMPRÍMASE.

Ing. Murphy Olympto Paiz Recinos

DECANO



Guatemala, noviembre de 2009

/cc
c.c. archivo.

AGRADECIMIENTOS A:

Compañeros de estudios

Por su compañía, apoyo y cooperación a lo largo de las muchas horas de estudio y momentos que compartimos esforzándonos por alcanzar la meta de estudios.

Escuela Superior de Arte – ESA

Especialmente a la Directora Inga. Claudia Liliana Gil, por su colaboración y apoyo en la realización del proyecto asignado.

Universidad de San Carlos de Guatemala

Casa de estudios que me brindó la oportunidad de forjarme como profesional y a la cual he tratado de devolver con este proyecto los conocimientos adquiridos.

ACTO QUE DEDICO A:

MI PADRE ETERNO Por su infinita misericordia, porque sin Él no podría obtener el logro de mi carrera, ya que es el dador de la sabiduría e inteligencia.

MIS PADRES José Augusto Colindres Rodríguez y Fidelia Estefana Solorzano de Colindres, por su apoyo incondicional, por ser quienes forjaron en mí los deseos de superación y ser una mujer luchadora por sus sueños.

MI AMADO Edson Mirglony Ramírez Caal, por su amor y deseos que culmine el logro de una meta más en mi vida.

MIS HERMANAS María Elizabeth Colindres Solorzano y Magda Isabel Colindres de Ruedas, por sus consejos y apoyo en la búsqueda de mi superación a nivel intelectual.

MIS ABUELITOS D.E.P. Ricardo Colindres y Vicenta Rodríguez Monzón, Adrian Solórzano Maldonado y María Ceferina López Camey, por su cariño y comprensión.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
GLOSARIO	VII
RESUMEN	XIII
OBJETIVOS	XV
INTRODUCCIÓN	XVII
1. INFORMACIÓN DEL PROYECTO	1
1.1. Antecedentes del proyecto	1
1.2. Descripción general del proyecto	4
1.3. Descripción general del marco teórico	5
2. MARCO TEÓRICO CONCEPTUAL	7
2.1. Definición técnica	7
2.1.1. Consideraciones especiales	7
2.1.1.1. Definición del ciclo de vida del <i>software</i>	7
2.1.1.1.1. Elementos de un ciclo de vida	8
2.1.2. Utilización de <i>software</i> libre	8
2.1.2.1. Definición de <i>software</i> libre	9
2.1.2.2. Clasificación de licencias de <i>software</i> libre	10
2.1.3. Utilización de <i>software</i> propietario.	12
2.2. Alcances y limitaciones	13
2.2.1. Enfoque del proyecto	13

2.3.	Sistema de Gestión de Base de Datos	13
2.4.	Herramienta de desarrollo	16
2.4.1.	Modelo entidad-relación	17
2.4.2.	Diagramador	18
2.4.3.	Herramienta para desarrollar el <i>software</i>	18
2.4.3.1.	<i>Delphi</i>	18
2.4.3.1.1.	Uso y variantes	19
2.4.3.1.2.	Componentes	20
3.	COMPONENTES DE SOFTWARE	21
3.1.	Metodología implementada	21
3.1.1.	Fase de definición	21
3.1.1.1.	Estudio de viabilidad	21
3.1.1.2.	Toma de requerimientos	21
3.1.1.2.1.	Funciones y limitaciones	22
3.1.2.	Fase de diseño	22
3.1.2.1.	Soluciones tecnológicas	22
3.1.2.1.1.	Diagrama conceptual de la solución	22
3.1.2.2.	Recursos	24
3.1.4.	Fase de construcción	24
3.1.4.1.	Configuraciones del sistema	24
3.1.4.1.1.	Configuración de cursos	24
3.1.4.2.	Mantenimientos	26

3.1.4.2.1. Horario	26
3.1.4.3. Procesos	28
3.1.4.3.1. Asignación de cursos	28
3.1.4.4. Generación de informes	30
3.1.4.4.1. Actas de cursos	30
3.1.5. Fase de mantenimiento y operación	31
3.1.5.1. Instalación de la solución del sistema	31
3.1.5.1.1. Base de datos	31
3.1.5.2. Capacitación del usuario	31
3.1.5.3. Ingreso de datos	31
3.1.5.4. Mantenimiento	31
4. COMPONENTES DE HARDWARE	33
4.1. Recursos de <i>hardware</i>	33
4.2. Descripciones técnicas	33
5. CASOS DE USO EN LA SOLUCIÓN IMPLEMENTADA	35
5.1. Inscripción - ingreso/reingreso de estudiantes	35
5.2. Proceso de asignación de cursos	38
5.3. Carga de notas	40
5.4. Impresión de certificación de cursos	43

6. LICENCIA DEL PROYECTO	45
CONCLUSIONES	47
RECOMENDACIONES	49
REFERENCIAS ELECTRÓNICAS	51
BIBLIOGRAFÍA	53

ÍNDICE DE ILUSTRACIONES

FIGURAS

1. Software libre vrs gratuito	10
2. Diagrama conceptual de la solución	23
3. Pantalla del catálogo de cursos	24
4. Pantalla de configuración de horarios	26
5. Pantalla de configuración de horarios – salón período	27
6. Pantalla de asignación de cursos	28
7. Pantalla de mensaje de no cumple con los requisitos	29
8. Boleta de asignación	29
9. Informe de actas finales	30
10. Proceso de inscripción – ingreso/reingreso de estudiantes	37
11. Proceso de asignación de cursos	39
12. Carga de notas	42
13. Impresión de certificación de cursos	44

GLOSARIO

ActiveX Data Objects (ADO)

Es uno de los mecanismos que usan los programas de computadoras para comunicarse con las bases de datos, darles órdenes y obtener resultados de ellas.

Con ADO, un programa puede leer, insertar, editar, o borrar, la información contenida en diferentes áreas de almacenamiento dentro de la base de datos llamadas tablas. Además, se puede manipular la propia base de datos para crear nuevas áreas para el almacenamiento de información (tablas), como también alterar o eliminar las ya existentes, entre otras cosas.

Fue desarrollado por *Microsoft* y es usado en ambientes *Windows* por lenguajes de programación como *Visual Basic*, *C++*, *Delphi* entre otros, como también en la Web mediante el uso de *Active Server Pages (ASP)* y el lenguaje *VBScript*.

Base de Datos

Una base de datos es un “almacén” que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente.

Cadena de valor

La cadena de valor categoriza las actividades que producen valor añadido en una organización.

Código fuente

El código fuente de un programa informático (o *software*) es un conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar dicho programa. Por tanto, en el código fuente de un programa está descrito por completo su funcionamiento.

Free Software Foundation

La Fundación para el Software Libre (FSF) está dedicada a eliminar las restricciones sobre la copia, redistribución, entendimiento, y modificación de programas de computadoras. Con este objeto, promueve el desarrollo y uso del software libre en todas las áreas de la computación.

Freeware

El término inglés *freeware* define un tipo de *software* de computadora que se distribuye sin costo y por tiempo ilimitado.

Informática

Es la disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales. También es definida como el procesamiento de la información en forma automática. Para esto los sistemas informáticos deben realizar las siguientes tres tareas básicas:

- Entrada: Captación de la información digital.
- Proceso: Tratamiento de la información.
- Salida: Transmisión de resultados binarios.

Ingeniería de *Software*

Es la disciplina dentro de la informática encargada de la creación de *software* de calidad.

Metodología

Se refiere a los métodos de investigación que se siguen para alcanzar una gama de objetivos en una ciencia. Método es el procedimiento para alcanzar los objetivos y la metodología es el estudio del método.

Open Database Connectivity (ODBC)

Es un estándar de acceso a Bases de Datos , el objetivo de *ODBC* es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (*DBMS* por sus siglas en inglés) almacene los datos, *ODBC* logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el *DBMS*, el propósito de esta capa es traducir las consultas de datos de la aplicación en comandos que el *DBMS* entienda. Para que esto funcione tanto la aplicación como el *DBMS* deben ser compatibles con *ODBC*, esto es que la aplicación debe ser capaz de producir comandos *ODBC* y el *DBMS* debe ser capaz de responder a ellos.

Programación visual

La programación visual brinda los conocimientos necesarios para diseñar y desarrollar aplicaciones con un entorno visual amigable y fácil de utilizar para el usuario.

Servidor de Archivos

Tipo de servidor en una red de ordenadores cuya función es permitir el acceso remoto a archivos almacenados en él o directamente accesibles por este. En principio, cualquier ordenador conectado a una red con un *software* apropiado, puede funcionar como servidor de archivos. Desde el punto de vista del cliente de un servidor de archivos, la localización de los archivos compartidos es transparente. O sea, normalmente no hay diferencias perceptibles si un archivo está almacenado en un servidor de archivos remoto o en el disco de la propia máquina.

Algunos protocolos comúnmente utilizados en servidores de archivos:

- SMB/CIFS (*Windows*, Samba en Unix)
- NFS (Unix).

Sistema operativo

Un Sistema Operativo es el *software* encargado de ejercer el control y coordinar el uso del hardware entre diferentes programas de aplicación y los diferentes usuarios. Es un administrador de los recursos de *hardware* del sistema.

SSH

SSH (*Secure SHell*) -intérprete de comandos seguro- es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red.

Topología de redes

En algunos casos se puede usar la palabra arquitectura en un sentido relajado para hablar a la vez de la disposición física del cableado y de cómo el protocolo considera dicho cableado.

RESUMEN

El objetivo de este trabajo es la elaboración de un Sistema de Información, que permita llevar el control de estudiantes, catedráticos con sus diferentes localizaciones (dirección, teléfono, e-mail, etcétera); catálogo de cursos que pertenecen a un pensum en las diferentes carreras a nivel de licenciatura que ofrece la Escuela Superior de Arte –ESA, de la Universidad de San Carlos de Guatemala.

La configuración de horarios para los diferentes ciclos, así como el ingreso de asignaciones, carga de notas finales de ciclo, para poder generar Actas de Cursos (Iniciales y finales).

Desde el sistema de Información el estudiante puede solicitar certificaciones de cursos Aprobados.

El presente informe describe conocimientos técnicos adquiridos durante la carrera de Ingeniería en Ciencias y Sistemas, para poder generar el software del Sistema de Información del “Control Académico de la Escuela Superior de Arte –ESA-, de la Universidad de San Carlos de Guatemala”.

OBJETIVOS

GENERALES:

1. Con la propuesta de este proyecto cubrir las necesidades que se tienen en las unidades académicas de la Universidad de San Carlos de Guatemala, con el fin de llevar el control de notas en forma sistematizada.
2. A partir de este proyecto se pueda tener una base para generar otras versiones de sistemas, ya que la propuesta este proyecto tiene una licencia tipo COPYLEFT, que permite la distribución, estudio y copia del sistema.

ESPECÍFICOS:

1. Llevar el control de estudiantes de las diferentes carreras que imparte la Escuela.
2. Llevar el control de los catedráticos que imparten / impartieron cursos en los diferentes horarios programados por la Escuela.
3. Permitir un control de los diferentes horarios para los cursos programados en los diferentes ciclos.
4. Realizar la asignación de cursos por alumno en los diferentes ciclos.
5. Que exista un mecanismo de des-asignación de cursos por el estudiante en un ciclo determinado.
6. Proporcionar un mecanismo para la carga / ingreso de notas de cursos en los diferentes ciclos, según el horario programado por la Escuela.

7. Tener un mecanismo que controle que el ingreso de la nota es válida, según los mínimos y máximos asignados en la ponderación de notas por curso y ciclo.
8. Tener el control de cursos asignados que sean válidos cumpliendo con los requisitos para asignación de dicho curso.
9. Generar actas de notas en los diferentes cursos y ciclos planificados.
10. Proporcionar listados de cursos aprobados al estudiante que lo requiera en los formularios diseñados por la Escuela de Arte.

INTRODUCCIÓN

La creación de la Escuela Superior de Arte – ESA, cumple con dos funciones [i]:

- I. Satisfacer la demanda estudiantil e
- II. Innovar la oferta de estudios al crear cuatro carreras en el campo del arte.

La Escuela Superior de Arte – ESA, inició sus actividades docentes en enero de 2008, con una población inicial de 61 estudiantes en las licenciaturas de:

- Licenciatura en Arte Dramático con especialidad en Actuación
- Licenciatura en Artes Visuales con especialidad en Pintura
- Licenciatura en Música.

Los títulos que ofrece la escuela son, en el grado de licenciado: los siguientes:

- Licenciado en Arte Dramático con especialización en Actuación / Dirección.
- Licenciado en Artes Visuales con especialización en Pintura / Escultura.
- Licenciado en Ballet Clásico y Coreografía / Danza Contemporánea y Coreografía.
- Licenciado en Música.

Para la implementación de este proyecto se utilizó el ciclo de vida del *software*, utilización de *software* libre y según la clasificación se implemento una licencia tipo *copyleft*, haciendo un aporte a la comunidad, en especial a las Escuelas/ Facultades de la Universidad de San Carlos de Guatemala que necesiten utilizar el sistema, para llevar el Control Académico.

En los sistemas de Gestión de Base de Datos (SGBD) el propósito es manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente

se convertirán en información relevante, para un buen manejo de datos. Dentro de los objetivos que debe de cumplir los SGBD son:

- Abstracción de la información.
- Independencia.
- Consistencia.
- Seguridad.
- Integridad.
- Respaldo.
- Control de la concurrencia.
- Manejo de Transacciones.
- Tiempo de respuesta.

Para implementar este proyecto se utilizó un Sistema de Gestión de Bases de Datos libre: *PostgreSQL* Licencia BSD y la Herramienta de Desarrollo de *Software* (HDS) : Delphi

Dentro de los componentes de software y la metodología implementada se puede mencionar:

- Fase de definición.
- Fase de diseño.
- Fase de construcción.
- Fase de Mantenimiento y operación.

Entre los diagramas como solución de software implementado se tienen diagramas de casos de uso de los procesos principales. Además se incluye la licencia del *software* desarrollado.

1. INFORMACIÓN DEL PROYECTO

1.1. Antecedentes del proyecto

En el año de 1985 se creó la primera comisión para el desarrollo del proyecto para formar la Escuela Superior de Arte de la Universidad de San Carlos de Guatemala, dicha comisión la conformaron 4 maestros: Marco Augusto Quiroa (Artes Visuales), Antonio Crespo Morales (Danza), Víctor Hugo Cruz (Arte Dramático) y Jorge Alvarado Sarmientos (Música); acompañando a los maestros por parte de la USAC el Lic. Luis Vallejo en la organización administrativa y al Ing. Jorge Cáceres como Coordinador General. En 1986 el proyecto queda inerte con la partida del Dr. Eduardo Meyer como Rector de la USAC. En 1994 se nombra al Arq. Lionel Méndez D'Avila quien asume como nuevo Coordinador General de la comisión, misma que hace entrega en 1998 del anteproyecto global, donde se establecen las fases de trabajo y procedimientos que seguirá la comisión para crear el proyecto final. El 11 de noviembre de 1999 por instrucciones del Rector Ing. Efraín Medina Guerra se integran a la comisión las licenciadas María Reinhardt y Elisa González de Marroquín Profesoras Titulares de la USAC, quienes actúan en calidad de curriculistas e investigadoras para conformar el estudio académico del proyecto. Asimismo colaboran con esta comisión el Maestro MM Igor Sarmientos, para apoyar la creación del pensum de música, la descripción y contenido de los cursos de esa carrera y la Inga. Claudia Gil, para apoyar el desarrollo de los estudios administrativo, financiero y legal. Integrándose a la comisión las licenciadas y profesionales antes mencionados en el año de 2002, ya que tenían varios años de estar apoyando y trabajando en el proyecto en forma

totalmente desinteresada y ad-honorem. Haciendo entrega del proyecto final en junio de 2002. Dicho proyecto fue aprobado en la sesión del 19 de abril de 2006.

El 16 de mayo de 2006, se emite el Acuerdo de Rectoría No. 602-2006, donde se amplía y modifica el Acuerdo de Rectoría No. 443-2002 para nombrar la Comisión de Acompañamiento para la Ejecución del Proyecto de la Escuela Superior de Arte.

Entre los fines que persigue la Escuela es:

- Elevar el nivel cultural del guatemalteco, a través de la creación artística.
- Formar artistas con capacidad de caracterizar lo guatemalteco y universalizarlo.
- Visualizar que la educación superior es esencial para que todo el país alcance el nivel necesario de desarrollo sostenible-natural a partir del patrimonio cultural.

Para alcanzar los fines de la escuela en el egresado de la escuela, se diseñaron cursos que cubren los ámbitos:

- Científico: Éste es el del –homo sapiens-. Está orientado a formar el pensamiento racional, lo integran las habilidades del pensamiento abstracto y formal, en correspondencia con lo cuantitativo. Esta formación ubica al estudiante en el mundo de las ciencias, al proporcionarle principios, leyes y criterios para obtener una representación real, objetiva y verificable del mundo ante los problemas y situaciones de su vida profesional.

- Social: Éste es el del –homo sapiens-. Está orientado a formar en el estudiante el pensamiento reflexivo y crítico. Esta formación ubica al estudiante en el análisis del origen y desarrollo de la sociedad y sus instrucciones, así como de las ideas y relaciones que configuran la vida social del hombre.

- Humanístico: Éste comprende a los del –homo esteticus y el del homo eticus-. Está orientado a formar el pensamiento creador, liberando todas las formas, según el constructivismo endógeno y exógeno. En correspondencia con lo cualitativo, esta formación ubica al estudiante en el análisis de la esencia constitutiva del hombre como ser y ente creador y lúdico.

- Tecnológico: Éste es el del –homo faber-. Está orientado a fortalecer el proceso intelectual, con énfasis en el dominio motriz. Esta formación ubica al estudiante en el saber hacer, definido como el conjunto de conocimientos propios de una técnica para formar un profesional hábil, con el máximo desarrollo de destrezas y habilidades, capaz de desenvolverse en el mundo práctico.

- De la especialidad artística: Éste comprende a los del –homo integrus y esteticus-. Está orientado a fortalecer la creación, con un pensamiento integrador y sintético, en donde lo racional y sentimental; lo teórico y lo práctico; lo llevan a la creación de una obra única, idiosincrática, irrepetible. Esta formación ubica al estudiante en el expresar su ser, su voluntad, a la autorrealización, a la alegría.

Este proyecto tiene como finalidad llevar el control de notas de los cursos aprobados por los estudiantes de las carreras que se imparten en la Escuela Superior de Arte.

1.2. Descripción general del proyecto

Llevar el control de alumnos inscritos y asignados en los ciclos definidos como:

- Primer/ Segundo Semestre
- 1ra. Retrasada (Primer/Segundo Semestre)
- 2da. Retrasada (Primer/Segundo Semestre)
- Exámenes de Suficiencia
- Privado y Público.

En asignaciones Regular o Extraordinario.

Así como el control de horarios, pensum de estudio por carrera, generación de boletas de asignación, carga de notas por período, generación de actas de notas, certificaciones de cursos, control de catedráticos, secciones en cursos especiales donde se tienen varios catedráticos asignados a una especialidad (por ejemplo: Instrumento I, con sus diferentes especialidades en Guitarra, Piano, Marimba etc.), en sí se estima que son 52 instrumentos con los que cuenta una orquesta.

Las áreas que cubre el pensum de estudios son:

- A – Artes
- AD – Arte Dramático
- AV – Artes Visuales
- D – Danza

- M – Música

Ámbitos son:

- C = Científico
- S = Social
- H = Humanístico
- T = Tecnológico
- EA = Especialidad Artística

Obligatoriedad:

- * = Obligatoria
- o = Optativa

Niveles

- I = Básico
- II = Profesional
- III = Integrado

Semestres

- 1 2 3 4 5 6 7 8 9 10.

1.3. Descripción general del marco teórico

La creación e implementación de un sistema de información es un proceso complejo que requiere de mucho conocimiento y bases teóricas en varios campos de la informática.

Dentro de los conocimientos y bases teóricas se consideraron los siguientes:

- Ingeniería de Software
 - ✓ Metodología del software aplicada en la solución del sistema de información.

Toma de Requerimientos: Obtener información sobre las necesidades que se quieren cubrir, al implementar el sistema de información.

Análisis y diseño: El análisis que es lo que se va a implementar, y en cuanto al diseño la forma en que se presentará la solución.

Implementación de la solución: La herramienta de desarrollo y el lenguaje de programación, arquitectura que se implementará en la aplicación.

- Redes de computadoras

Topología de red implementada en la solución del sistema de información.

- Sistemas operativos

Se aplicó un sistema operativo de uso libre en el servidor de la base de Datos y servidor de Archivos.

- Sistema de gestión de bases de datos

Como parte de la solución en la implementación del sistema de información se realizó en una base de datos de uso libre.

2. MARCO TEÓRICO CONCEPTUAL

2.1. Definición técnica

En el presente trabajo de graduación, el objetivo fundamental del proyecto fue realizar un sistema de información para la Escuela Superior de Arte – ESA de la Universidad de San Carlos de Guatemala.

Dentro de los objetivos indicados en el presente trabajo, es la agilización de la información requerida por control académico de dicha entidad.

2.1.1. Consideraciones especiales

Para el diseño y construcción de agregados al sistema de información que cumpla con los objetivos del proyecto y se obtuviera como resultado el incremento a la cadena del valor al sistema de información presentado en este proyecto, se debe de aplicar como parte de la solución la implementación del ciclo de vida del software¹ que mejor se ajusto a las necesidades de implementación.

2.1.1.1. Definición del ciclo de vida del *software*

Todo proyecto de ingeniería tiene fines ligados a la obtención de un producto, proceso o servicio que es necesario generar a través de diversas actividades.

¹ **Ciclo de Vida del Software:** Un modelo de ciclo de vida define el estado de las fases a través de las cuales se mueve un proyecto de desarrollo de software.

Algunas de estas actividades pueden agruparse en fases porque globalmente contribuyen a obtener un producto intermedio, necesario para continuar hacia el producto final y facilitar la gestión del proyecto. Al conjunto de las fases empleadas se le denomina “ciclo de vida”. [2]

2.1.1.1. Elementos de un ciclo de vida

Un ciclo de vida para un proyecto se compone de **fases sucesivas** compuestas por tareas planificables. Según el modelo de ciclo de vida, la sucesión de fases puede ampliarse con **bucles de realimentación**, de manera que lo que conceptualmente se considera una misma fase se pueda ejecutar más de una vez a lo largo de un proyecto, recibiendo en cada pasada de ejecución aportaciones de los resultados intermedios que se van produciendo (realimentación). [3]

2.1.2. Utilización de *software* libre

La utilización de herramientas de software libre "*free software*"², para la creación y puesta en funcionamiento del sistema no es objetivo propio del sistema de información. [4]

² **Software libre** (en inglés *free software*).

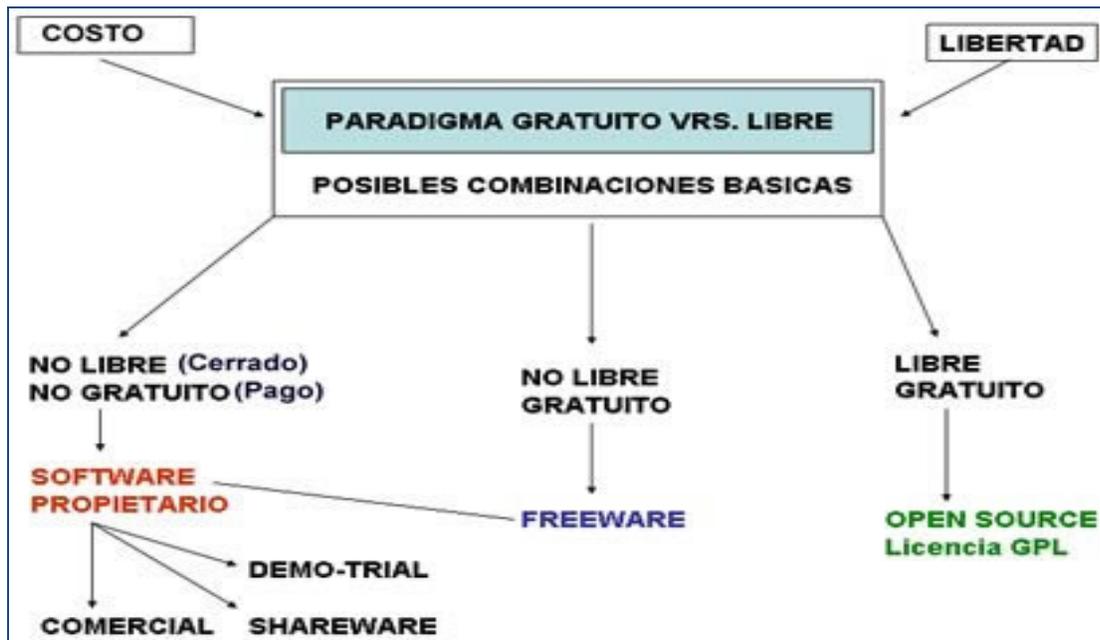
2.1.2.1. Definición de *software* libre

Según la *Free Software Foundation*, el *software* libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el *software*; de modo más preciso, se refiere a cuatro libertades de los usuarios del *software*:

- Libertad de usar el programa, con cualquier propósito.
- Estudiar el funcionamiento del programa, y adaptarlo a las necesidades.
- Distribuir copias, con lo que puede ayudar a otros.
- Mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie (para la segunda y última libertad mencionadas, el acceso al código fuente es un requisito previo).

El *software* libre suele estar disponible gratuitamente, o al precio de costo de distribución a través de otros medios; sin embargo no es obligatorio que sea así, por ende no hay que asociar *software* libre a "*software* gratuito" (denominado usualmente *freeware*), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("*software* comercial"). Análogamente, el "*software* gratis" o "gratuito" incluye en algunas ocasiones el código fuente; no obstante, este tipo de *software* no es libre en el mismo sentido que el *software* libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa, como lo muestra la siguiente figura:

Figura 1. *Software* libre vrs gratuito



Tampoco debe confundirse *software* libre con "*software* de dominio público". Éste último es aquél que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, porque pertenece a todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este *software* sería aquél cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado, tras un plazo contado desde la muerte de éste, habitualmente 70 años. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es dominio público.

2.1.2.2. Clasificación de licencias de *software* libre

Es posible dividir las licencias de *software* libre en dos grandes familias.

- Una de ellas está compuesta por las licencias que no imponen condiciones especiales, sólo especifican que el *software* se puede redistribuir o modificar. Estas son las llamadas licencias permisivas.
- La otra familia, denominadas licencias robustas o licencias *copyleft*³, imponen condiciones en caso de que se quiera redistribuir el *software*, condiciones que van en la línea de forzar a que se sigan cumpliendo las condiciones de la licencia después de la primera redistribución.

Mientras que el primer grupo hace énfasis en la libertad de quien recibe un programa, ya que le permite hacer casi lo que quiera con él (en términos de las sucesivas redistribuciones), el segundo obliga a que las modificaciones y redistribuciones respeten los términos de la licencia original. [5]

Económicamente, el *copyleft* se considera el único mecanismo capaz de competir con las empresas monopolísticas que dependen de la explotación económica del *copyright*, marcas registradas y leyes de patentes. El *copyleft* permite a programadores voluntarios contribuir y sentirse involucrados en el desarrollo de software, formando parte de un proyecto más grande, como por ejemplo el desarrollo de un sistema de información (Control Académico). Además, se aseguran de que cualquier derivado que surja de su esfuerzo en el futuro permanecerá accesible a ellos gracias al *copyleft*. Por consiguiente, el desarrollo de *software* con *copyleft* deja clara su intención de nunca ocultar o abusar de cualquier conocimiento que se aporte. A su vez, el *copyleft* también garantiza que las compañías y programadores que se deciden a colaborar no puedan crear sus propias versiones privativas del trabajo para tomar ventaja

³ **Copyleft** o *copia permitida* (=left(de leave) =granted).

sobre otros. En su lugar, la competitividad se basa en otros aspectos del suministro de productos comerciales con *copyleft*. [6]

2.1.3. Utilización de *software* propietario.

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

Para la Fundación para el *Software* Libre (FSF) este concepto se aplica a cualquier *software* que no es libre o que sólo lo es parcialmente (semi libre), sea porque su uso, redistribución o modificación está prohibida, o requiere permiso expreso del titular del *software*.

En el *software* no libre una persona física o jurídica (compañía, corporación, fundación, etc.) posee los derechos de autor sobre un *software* negando o no otorgando, al mismo tiempo, los derechos de usar el programa con cualquier propósito; de estudiar cómo funciona el programa y adaptarlo a las propias necesidades (donde el acceso al código fuente es una condición previa); de distribuir copias; o de mejorar el programa y hacer públicas las mejoras (para esto el acceso al código fuente es un requisito previo).

De esta manera, un *software* sigue siendo no libre aún si el código fuente es hecho público, cuando se mantiene la reserva de derechos sobre el uso,

modificación o distribución (por ejemplo, la versión comercial de SSH o el programa de licencias *shared source de Microsoft*). [7]

2.2. Alcances y limitaciones

Dentro de sus alcances de este proyecto es la carga de notas en forma automática por medio de archivos de Excel; el sistema hace el formato con el carné, nombre del estudiante, nota de laboratorio, zona, nota de examen final y nota final.

La limitación es que es una aplicación de escritorio.

2.2.1. Enfoque del proyecto

Este proyecto se enfoca en un tipo de licencia *COPYLEFT*, pues esto hace un aporte a la comunidad, en especial a las escuelas/ facultades de la Universidad de San Carlos de Guatemala que necesiten utilizar el sistema, para llevar el Control Académico.

2.3. Sistema de Gestión de Base de Datos

Los Sistemas de Gestión de Base de Datos (SGBD) son un tipo de *software* muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

El propósito general de los Sistemas de Gestión de Base de Datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante, para un buen manejo de datos.

Existen distintos objetivos que deben cumplir los SGBD:

- **Abstracción de la información.** Los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario. Así, se definen varios niveles de abstracción.
- **Independencia.** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.
- **Consistencia.** En aquellos casos en los que no se ha logrado eliminar la redundancia, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea. Por otra parte, la base de datos representa una realidad determinada que tiene determinadas condiciones, por ejemplo que los menores de edad no pueden tener licencia de conducir. El sistema no debería aceptar datos de un conductor menor de edad. En los SGBD existen herramientas que facilitan la programación de este tipo de condiciones.

- **Seguridad.** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra segura frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.
- **Integridad.** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de *hardware*, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada. Los SGBD proveen mecanismos para garantizar la recuperación de la base de datos hasta un estado consistente.
- **Respaldo.** Los SGBD deben proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.
- **Control de la concurrencia.** En la mayoría de entornos (excepto quizás el doméstico), lo más habitual es que sean muchas las personas que acceden a una base de datos, bien para recuperar información, bien para almacenarla. Y es también frecuente que dichos accesos se realicen de

forma simultánea. Así pues, un SGBD debe controlar este acceso concurrente a la información, que podría derivar en inconsistencias.

- **Manejo de Transacciones.** Una Transacción es un programa que se ejecuta como una sola operación. Esto quiere decir que el estado luego de una ejecución en la que se produce una falla es el mismo que se obtendría si el programa no se hubiera ejecutado. Los SGBD proveen mecanismos para programar las modificaciones de los datos de una forma mucho más simple que si no se dispusiera de ellos.
- **Tiempo de respuesta.** Lógicamente, es deseable minimizar el tiempo que el SGBD tarda en darnos la información solicitada y en almacenar los cambios realizados⁴.

Para implementar este proyecto se utilizó un Sistema de Gestión de Bases de Datos (SGBD) libre: *PostgreSQL*⁵ Licencia BSD.⁶

2.4. Herramienta de desarrollo

Las herramientas de desarrollo de *software* (HDS) han desempeñado un importante papel en el desarrollo de aplicaciones. Como consecuencia del avance tecnológico éstas han experimentado también continuos cambios. Así como se cuenta en la actualidad con documentación sobre las numerosas HDS

⁴ En inglés: *Database management system*, abreviado **DBMS**.
http://es.wikipedia.org/wiki/Sistema_de_gesti3n_de_base_de_datos

⁵ www.postgresql.org

⁶ Pertenece al grupo de licencias de software Libre.

disponibles, y con trabajos de investigación que revelan avances en herramientas particulares, existe también en la literatura evidencia de que los escritos técnicos genéricos sobre HDS son relativamente escasos, identificándose la pertinencia de canalizar iniciativas para la formulación de modelos conceptuales, que apoyen la comprensión (y el uso posterior) de las HDS. [9]

2.4.1. Modelo entidad-relación

El Modelo Entidad-Relación es un concepto de modelado para bases de datos, propuesto por Peter Chen en 1976, mediante el cual se pretende 'visualizar' los objetos que pertenecen a la Base de Datos como entidades (esto es similar al modelo de Programación Orientada a Objetos - POO) las cuales tienen unos atributos y se vinculan mediante relaciones.

Es una representación conceptual de la información. Mediante una serie de procedimientos se puede pasar del modelo E-R a otros, como por ejemplo el modelo relacional.

El modelado entidad-relación es una técnica para el modelado de datos utilizando diagramas entidad relación. No es la única técnica pero sí la más utilizada. Brevemente consiste en los siguientes pasos:

1. Se parte de una descripción textual del problema o sistema de información a automatizar (los requisitos).
2. Se hace una lista de los sustantivos y verbos que aparecen.
3. Los sustantivos son posibles entidades o atributos.
4. Los verbos son posibles relaciones.

5. Analizando las frases se determina la cardinalidad de las relaciones y otros detalles.
6. Se elabora el diagrama (o diagramas) entidad-relación.
7. Se completa el modelo con listas de atributos y una descripción de otras restricciones que no se pueden reflejar en el diagrama.

Dado lo rudimentario de esta técnica se necesita cierto entrenamiento y experiencia para lograr buenos modelos de datos. [10]

2.4.2. Diagramador

Para realizar el diagrama entidad-relación del proyecto se utilizó una herramienta versión *Shareware* de *Case Studio 2*.

Es una herramienta de diseño de base de datos con las siguientes características:

- Fácil acceso a cualquier base de datos relacional.
- Comparación comprensiva entre el modelo de datos y la base de datos.
- Soporta la separación del modelo lógico y del físico.

2.4.3. Herramienta para desarrollar el *software*

2.4.3.1. *Delphi*

Delphi es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual. En *Delphi* se utiliza como lenguaje de programación una versión moderna de *Pascal* llamada *Object*

Pascal. En sus diferentes variantes, permite producir archivos ejecutables para *Windows*, *Linux* y la plataforma .NET. [11]

2.4.3.1.1. Uso y variantes

Un uso habitual de *Delphi* (aunque no el único) es el desarrollo de aplicaciones visuales y de bases de datos cliente-servidor y multicapas. Debido a que es una herramienta de propósito múltiple, se usa también para proyectos de casi cualquier tipo, incluyendo aplicaciones de consola, aplicaciones de *web* (por ejemplo servicios *web*, CGI, ISAPI, NSAPI, módulos para Apache), servicios COM y DCOM, y servicios del sistema operativo. Entre las aplicaciones más populares actualmente destaca *Skype*, un programa de telefonía por IP.

Delphi inicialmente sólo producía ejecutables binarios para *Windows*: *Delphi 1* para Win16 y con *Delphi 2* se introdujo Win32. En la actualidad da más posibilidades:

- En la última versión *Developer Studio 2006* incluye en el mismo entorno de desarrollo los lenguajes:
 - *Delphi* para Win32
 - *Delphi* para .NET
 - C# para .NET
 - C++
- Existe una versión de *Delphi* para sistemas *Unix* y *Linux*, denominada *Kylix* (de la cual existe una versión gratuita, aunque limitada).

2.4.3.1.2. Componentes

Delphi dio una implementación muy buena a la idea del uso de componentes, que son piezas reutilizables de código (clases) que pueden interactuar con el Entorno Integrado de Desarrollo (EID) en tiempo de diseño y desempeñar una función específica en tiempo de ejecución. Desde un enfoque más específico de la herramienta, se catalogan como componentes todos aquellos objetos que heredan de la clase *TComponent*, donde se implementa la funcionalidad necesaria para interactuar con el entorno de desarrollo, la carga dinámica desde *streams* y la liberación de memoria mediante una jerarquía. Una gran parte de los componentes disponibles para *Delphi* son *controles* (derivados de *TControl*), que encapsulan los elementos de interacción con el usuario como botones, menús, barras de desplazamiento, etcétera.

Delphi incluye una biblioteca de clases bien diseñada denominada VCL (*Visual Component Library*, Biblioteca de Componentes Visuales) y, en sus versiones 6 y 7, una jerarquía multiplataforma paralela denominada CLX. Ésta también se incluye en *Kylix*. Estas jerarquías de objetos incluyen componentes visuales y no visuales, tales como los pertenecientes a la categoría de acceso a datos, con los que puede establecerse conexiones de forma nativa o mediante capas intermedias (como ADO, BDE u ODBC) a la mayoría de las bases de datos relacionales existentes en el mercado. La VCL también está disponible para el desarrollo en .NET.

Además de poder utilizar en un programa estos componentes estándar (botones, grillas, conjuntos de datos, etc.), es posible crear nuevos componentes o mejorar los ya existentes, extendiendo la funcionalidad de la herramienta. En Internet existe un gran número de componentes, tanto gratuitos como comerciales, disponibles para los proyectos a los que no les basten los que vienen ya con la herramienta.

3. COMPONENTES DE SOFTWARE

3.1. Metodología implementada

A continuación se describen las tareas que fueron implementadas en la solución del Sistema de Información.

3.1.1. Fase de definición

En esta fase se estudió el propósito de la creación del sistema de Control Académico de la Escuela Superior de Arte.

3.1.1.1. Estudio de viabilidad

La Escuela Superior de Arte contó con un presupuesto destinado para compra del equipo de cómputo, para la realización del proyecto.

En cuanto al presupuesto del desarrollo del software, se estimó con base a un salario promedio para los profesionales de ingeniería en ciencias y sistemas como requisito para realizar el proyecto de EPS.

3.1.1.2. Toma de requerimientos

Se hicieron visitas periódicas durante aproximadamente 2 meses para la toma de requerimientos. Estos requerimientos fueron: La codificación de cursos, el pensum de estudios, el control de notas.

3.1.1.2.1. Funciones y limitaciones

El sistema de Información, trabaja en un ambiente de ventanas amigable para el usuario. Se estandarizó el uso de accesos directos en los botones asociados a eventos como por ejemplo: Creación de nuevos registros, guardar modificaciones, Impresión de documentos.

Una de las limitaciones que presenta el presente proyecto es el hecho que no fue desarrollado en ambiente *web*.

3.1.2. Fase de diseño

3.1.2.1. Soluciones tecnológicas

3.1.2.1.1. Diagrama conceptual de la solución

La solución implementada a nivel de usuario es la siguiente:

- ✓ Configuraciones del Sistema
 - Cursos
 - Período
 - Salón
- ✓ Pensum de Estudios
 - Pensum por Carrera
- ✓ Datos Generales
 - Estudiante
 - Catedrático
- ✓ Horarios
 - Horarios por ciclo y año
- ✓ Asignaciones y Control de Notas
 - Proceso de asignación

- Carga de Notas (Manual / Automática)
- Proceso de des-asignación
- ✓ Informes
 - Actas de notas
 - Certificaciones de cursos
 - Consulta de notas
 - Listado de alumnos por curso

Figura 2. Diagrama conceptual de la solución.



3.1.2.2. Recursos

3.1.4. Fase de construcción

3.1.4.1. Configuraciones del sistema

3.1.4.1.1. Configuración de cursos

El código del curso es generado por el sistema, a partir de las configuraciones de la pantalla por medio del ingreso de la información (cajas de diálogo, combos, etc.)

Figura 3. Pantalla del catálogo de cursos.

Curso	Código	Nombre	Ámbito	Nivel	Ob.	Sem.	Créd.
1	ACI 1.1*	Metodología de la investigación	Científico	Básico	1	1	3
2	ASI 1.1o	Semiótica y arte	Social	Básico	0	1	3
3	AHI 1.1*	Fenomenología de las artes I	Humanístico	Básico	1	1	2
4	ADEAI 1.1*	Teoría y práctica de la actuación I	Especialidad A	Básico	1	1	5
5	ADEAI 1.2*	Teoría y técnica de la dirección I	Especialidad A	Básico	1	1	5
6	ADTI 1.1*	Movimiento creativo	Tecnológico	Básico	1	1	5
7	ADTI 1.2*	Ortofona y técnica vocal I	Tecnológico	Básico	1	1	5
8	AVCI 1.1*	Geometría I	Científico	Básico	1	1	5
9	AVEAI 1.1*	Educación visual I	Especialidad A	Básico	1	1	5
10	AVSI 1.2*	Cultura I	Social	Básico	1	1	2
11	AVSI 1.3o	Cultura maya: arquitectura	Social	Básico	0	1	3
12	AVTI 1.1*	Expresión gráfica I	Tecnológico	Básico	1	1	10
13	D/ADSI 1.2o	Danzas populares regionales: occidente de Guatemala	Social	Básico	0	1	3
14	DEAI 1.1*	Acondicionamiento corporal	Especialidad A	Básico	1	1	5
15	DEAI 1.2*	Música I	Especialidad A	Básico	1	1	2

Listo Operador: 0 - [gecolindres] Fecha: 15-ago-2008

1. **Ámbito**

Se puede definir en los siguientes ámbitos

- Científico - **C**
- Social - **S**
- Humanístico - **H**
- Tecnológico - **T**
- Especialidad Artística – **EA**

2. **Nivel**

Se definieron los siguientes niveles

- Básico - **I**
- Profesional - **II**
- Integrado – **III**

3. **Área**

Se pueden seleccionar una o varias áreas a las que pertenece el curso.

Por ejemplo el curso:

Código: **D/ADSI1.2o**

Nombre: *Danzas populares regionales: occidente de Guatemala.*

Pertenece a las áreas de: Arte Dramático y Danza.

El código del curso es generado de la siguiente manera:

Área + Ámbito + Nivel + semestre + . + correlativo+ obligatorio

Donde el correlativo es la combinación de Área+Ámbito+Nivel, no hace referencia al semestre.

3.1.4.2. Mantenimientos

En estas pantallas se ingresa la información de estudiantes, catedráticos, configuraciones de horarios.

3.1.4.2.1. Horario

En esta ventana se define el horario que le corresponde a cada ciclo.

Por ejemplo: como lo muestra la figura 4, el horario 7 pertenece a la primera retrasada del primer semestre del año 2009.

Figura 4. Pantalla de configuración de horarios

Escuela Superior de Arte **Horario**

Año: 2009 T. Horario: Regular Ciclo: 1ra. Retrasada (1er. Semestre) - [2]

Horario: 7

Observaciones:

Curso: MEAI 2.2* Armonia I

Sección: Única Fecha:

Catedrático: ERNESTO EUGENIO CALDERON

Nota Mínima		Nota Máxima	
Laboratorio	0	Laboratorio	0
Zona	31	Zona	70
		Examen Final	30
		Nota	100

Curso: Salón-Periodo

Listo | Operador: 0 - [gecolindres] | Fecha: 14-jun-2009

Encabezado

Detalle

- **Encabezado**

Se ingresa el nuevo horario que pertenece a un ciclo.

- **Detalle**

El detalle de la ventana tiene dos tarjetas para definir el horario:

Curso: Se define el curso, la sección, el catedrático que imparte el curso, si el horario es para retrasada se ingresa la fecha de la retrasada.

Luego en la sección de la nota mínima y máxima del curso para dar un total de 100 puntos. Esta configuración es la base cuando se cargue las notas del ciclo y verificar si el estudiante aprueba el curso.

Salón-Período: Se define el salón o salones en que se imparte el curso, el período y los días asociados.

Figura 5. Pantalla de configuración de horarios - salón período

The screenshot shows a software interface for configuring schedules. At the top, there are navigation arrows and course information: "Curso: AVSI 1.2* Cultura I" and "Sección: Única". Below this, there are several icons representing different actions. The main area is divided into three sections: "Salón", "Período", and "Día".

Salón	Período
Catedra I	08:00 - 09:00

Salón	Período
Catedra I	08:00 - 09:00

Día
<input type="checkbox"/> Lunes - 1
<input checked="" type="checkbox"/> Martes - 2
<input type="checkbox"/> Miércoles - 3
<input checked="" type="checkbox"/> Jueves - 4
<input type="checkbox"/> Viernes - 5
<input type="checkbox"/> Sábado - 6

At the bottom, there are two tabs: "Curso" and "Salón-Período", with "Salón-Período" being the active tab.

Como lo muestra el ejemplo de la Figura 5, para el curso es:

Curso: Cultura I
Sección: Única (Si solo es una sección).
Salón: Cátedra I
Período: 8:00 – 9:00
Día: Martes y Jueves

3.1.4.3. Procesos

En estas pantallas se ingresa la información de asignación de cursos, carga de notas, des-asignación de cursos.

3.1.4.3.1. Asignación de cursos

Al realizar el proceso de asignación se puede fijar el área de la ventana (hacer clic para activar/desactivar) donde se define el año, tipo de asignación y el ciclo, para no estar configurando esta información con cada asignación por estudiante.

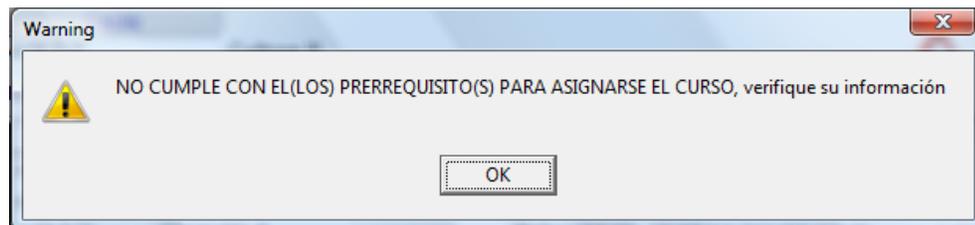
Figura 6. Pantalla de asignación de cursos



Con **F3**, se hace la búsqueda del estudiante que se asignará. Si el estudiante ya fue asignado para el año y ciclo correspondiente, se activa la información de la asignación. De lo contrario se debe de ingresar los cursos correspondientes a la asignación.

El sistema revisa los prerrequisitos cuando se está asignando un curso (para activar el listado de cursos **F3** en la caja de diálogo de Curso), si no cumple con los requisitos muestra el siguiente mensaje:

Figura 7. Pantalla de mensaje que no cumple con los requisitos



Luego de terminado el proceso de ingreso de cursos, se genera una boleta de asignación como la siguiente:

Figura 8. Boleta de asignación

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
Escuela Superior de Arte - ESA

BOLETA DE ASIGNACIÓN 200945-42-220120091254
2008 - 2da. Retrasada (2do. Semestre) [6]



Estudiante	
Carné	199716119
Nombre	LUIS EMILIO ROBLES GATICA
Carrera	Licenciatura en Artes Visuales con especialidad en Pintura
Código	Curso
ASI 2.1o	Antropología aplicada
	Catedra I [08:00 - 09:00] --- mie ---
AHI 2.1*	Fenomenología de las artes II
	Catedra I [08:00 - 09:00] --- mie ---

14-jun-2009 08:23 p.m. [0] gecolindres

La boleta de asignación se imprime en un formato media carta con copia. La original firmada y sellada por control académico se le entrega al estudiante y la copia firmada por el estudiante se archiva en control académico.

3.1.4.4. Generación de Informes

3.1.4.4.1. Actas de cursos

Estas son las actas que se archivan en Control Académico, las actas iniciales que no tiene nota asignada se pueden tomar como listado de alumnos por curso. Las actas finales son como lo muestra la Figura 9 a continuación.

Figura 9. Informe de Actas Finales

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA Escuela Superior de Arte - ESA		Acta de fin de curso			
Curso	AHI 1.1* - Fenomenología de las artes I	Año : 2008			
Horario	Catedra I (09:00 - 10:00) --- mar --- jue --- ---	Ciclo : 1 - 1er. Semestre			
Catedrático	66 - DANIEL ALARCON				
Carné	Nombre	Laboratorio	Zona	Examen Final	Nota Final
200719035	GUSTAVO ANTONIO BRACAMONTE BAUTISTA	0	23	SDE	23
200721411	JESSE DAVID CARRERA GALICIA	0	42	NSE	42
200722116	AXEL CARLOS MEZA GUZMAN	0	40	22	62
200722151	DAMARIS ADELY QUECHE YACH	0	40	NSE	40
200810442	GERBER BAUDILO DEL CID ISTUPE	0	41	20	61
200810456	FABIOLA AGUIRRE FLORES	0	47	16	63
200810497	ANDREA JIMENA SOTO COLCHIN	0	48	22	70
200810509	LUIS FERNANDO ALFARO TORTOLA	0	20	SDE	20
200810532	BETZY JOSEFINA SIMAJ TAPAZ	0	46	28	74
200810548	JOSE GUILLERMO HERNANDEZ CORDERO	0	0	SDE	0
200810553	RODRIGO JAVIER MALDONADO ORTEGA	0	40	28	68
200810585	JONATHAN ISAAC ARDON ESPAÑA	0	52	15	67
200816706	LUIS FERNANDO SOTO CONTRERAS	0	40	21	61

15-ago-2008 03:23 p.m.(0) gecolindres

Página No 2

(F) _____ DANIEL ALARCON

3.1.5. Fase de mantenimiento y operación

3.1.5.1. Instalación de la solución del sistema

3.1.5.1.1. Base de datos

La base de datos *Postgres* 8.2 se instaló en el servidor con un sistema operativo *Linux Red Hat* 9.

3.1.5.2. Capacitación del usuario

Se capacitó al usuario sobre el uso del sistema, y se apoyo con la presencia del desarrollador de *Software* (el especialista), en los procesos de asignación de cursos, carga de notas, generación de actas de notas de inicio y fin de semestre, asignación de retrasadas.

3.1.5.3. Ingreso de datos

Ingreso de la información requerida al sistema para el control de asignación de cursos de primer ingreso y reingreso.

3.1.5.4. Mantenimiento

Se le dio soporte al software por un tiempo aproximado de 6 meses.

4. COMPONENTES DE *HARDWARE*

4.1. Recursos de *hardware*

El equipo instalado es el siguiente:

- 1 Servidor de base de datos (*Linux*)
- 1 Cliente (*Windows XP*)
- 1 Impresora Matricial de carro ancho.

4.2. Descripciones técnicas

Servidor	2 procesadores Dual Core Intel, 2x2 MB Cache, Memoria de 4 GB 533 MHz, controlador SATA con soporte para 2 discos duros, 2 discos duros de 160 GB, Drive CD-RW/DVD, controlador de video PCI ATI ES1000 a 33 MHz con 16 MB de memoria gráfica, Monitor de 17"
Cliente	Procesador Pentium de 4 GHZ, Memoria de 1024 MB DDR, Disco duro de 120 GB, Monitor de 19" pantalla plana LCD
UPS	500VA, con regulador de voltaje integrado
Impresora	Matriz de Punto de 24 agujas, Alta velocidad de borrador 529 cps, Buffer de 128K.

5. CASOS DE USO EN LA SOLUCIÓN IMPLEMENTADA

5.1. Inscripción - ingreso/reingreso de estudiantes

ID caso de uso: 1

Nombre caso de uso: INSCRIPCIÓN – ingreso/reingreso de estudiantes

Actores: Estudiante, Control Académico

Descripción:

1. El estudiante llena la boleta de pre-Inscripción con los datos generales: Nombre completo, carrera, número de carné, fecha de nacimiento, localizaciones (dirección, teléfono casa, celular, e-mail).
2. Control Académico ingresa los datos generales del estudiante.
3. Asigna en forma automática los cursos para el primer semestre.
4. Genera la boleta de Asignación de Cursos.
5. El estudiante firma de recibida la boleta de asignación.

Función en base de

datos: caf_asignacionauto

Pre-condiciones:

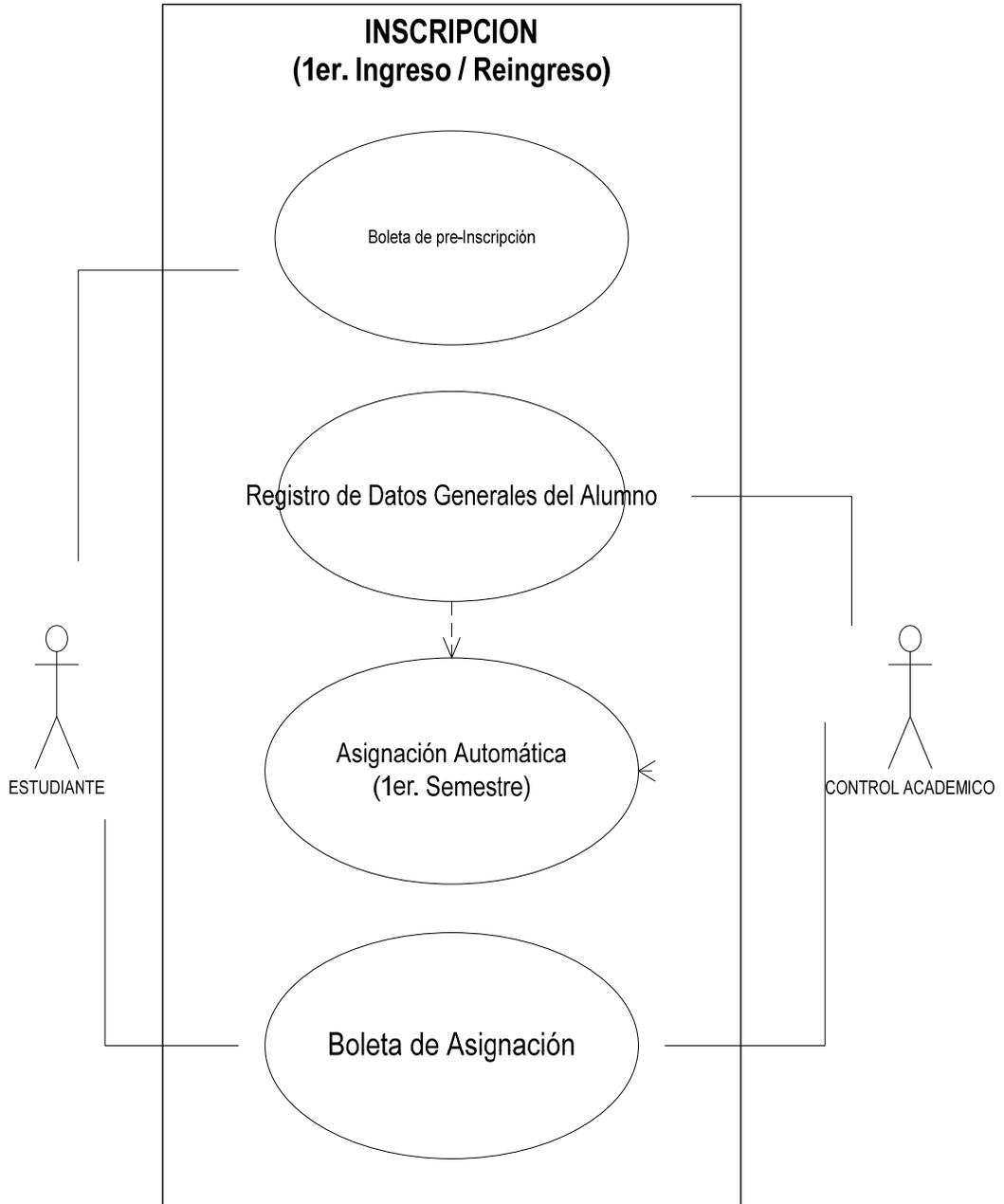
1. Estar inscrito en Registro y Estadística (Carné universitario).
2. Pago de inscripción para el ciclo actual.

- Flujo normal:**
1. Ingreso de los datos (estudiante inscrito, y pago de la cuota).
 2. Se realiza la asignación automática.
 3. Genera la boleta de cursos asignados.

Flujo alternativo: Para los estudiantes de música, la asignación del Instrumento Secundario es a través de la pantalla de de Asignación.

Frecuencia de uso:
Cada año

Figura 10. Proceso de Inscripción - ingreso/reingreso de estudiantes



5.2. Proceso de asignación de cursos

ID caso de uso: 2

Nombre caso de uso: ASIGNACIÓN DE CURSOS

Actores: Estudiante, Control Académico

Descripción:

1. El estudiante llena la boleta de Pre-Asignación (Asignación de Semestre/ Retrasadas/ Examen de Suficiencia) con los cursos que desea asignarse.
2. Control Académico realiza el proceso de asignación (En dicho proceso se verifica que el curso se pueda asignar, tomando en cuenta los prerrequisitos).
3. Genera la boleta de Asignación de Cursos.
4. El estudiante firma de recibida la boleta de asignación.

Pre-condiciones:

1. Para asignaciones del primer ciclo y estudiantes de reingreso, estar inscritos en el año de asignación de cursos.

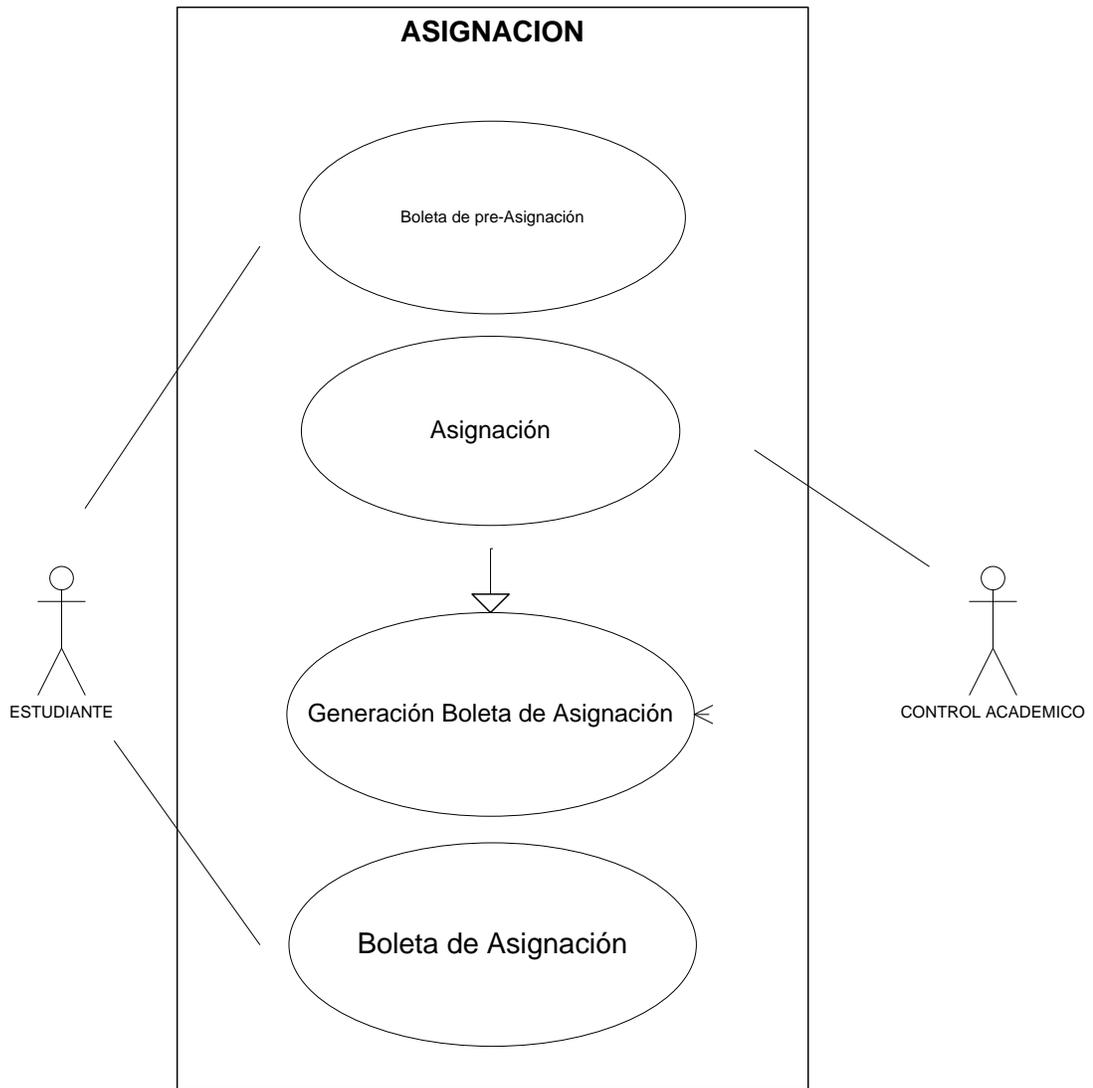
Flujo normal:

1. Ingreso de la asignación de cursos.
2. Generación de boletas de asignación.

Frecuencia de uso:

Cada vez que hay asignaciones de cursos.

Figura 11. Proceso de asignación de cursos



5.3. Carga de notas

ID Caso de uso: 3

Nombre caso de uso: CARGA DE NOTAS

Actores: Catedrático, Control Académico

Descripción:

1. Control Académico genera archivos con extensión csv (*Comma Separed Variables*), con el listado de alumnos asignados en el curso (carné, nombre, nota de laboratorio, zona, examen final y nota final).
2. El Catedrático llena el archivo generado en Excel las columnas: nota de laboratorio, zona, Examen final y nota final.
3. Control Académico puede hace la carga de notas en dos formas:
 - a. Forma anual: Cargan las notas que están en el acta de notas inicial.
 - b. Forma automática: Por medio del archivo *.csv, el sistema checa los controles para ver si la nota es válida.
4. Control Académico genera las actas de notas finales.

Función en base de datos: caf_cargaNota

Pre-condiciones: 1. Estar asignado en el curso que se está realizando la

carga de notas.

2. Que no esté des-asignado el curso.

Flujo normal:

1. Genera el archivo base con el listado de estudiantes asignados en el curso.

2. Se llenan las casillas correspondientes al laboratorio, zona, examen final y nota final.

3. Se hace la carga automática de notas.

4. Se generan las actas de Notas.

Flujo alternativo:

1. Se pueden generar las actas de notas iniciales (sólo para tener el listado de alumnos asignados en el curso).

2. Llena la información en las casillas de laboratorio, zona, examen final y nota final.

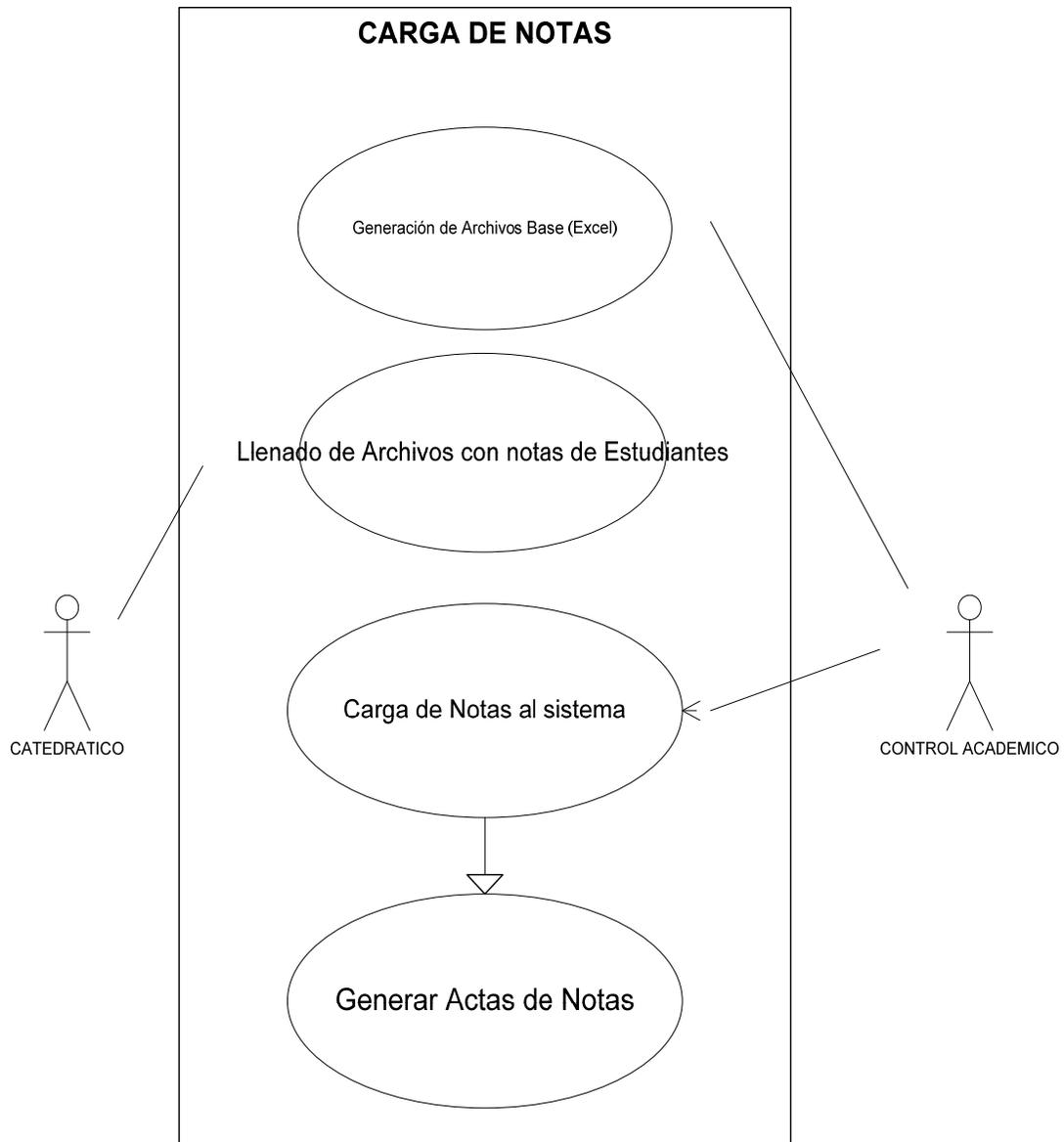
3. Se ingresan las notas en forma manual por Control Académico.

4. Se generan las actas de Notas.

Frecuencia de uso:

Cada vez que existan asignaciones de cursos se genera el proceso de carga de Nota.

Figura 12. Carga de notas



5.4. Impresión de certificación de cursos

ID Caso de uso: 4

Nombre caso de uso: IMPRESIÓN DE CERTIFICACIONES

Actores: Estudiante, Control Académico

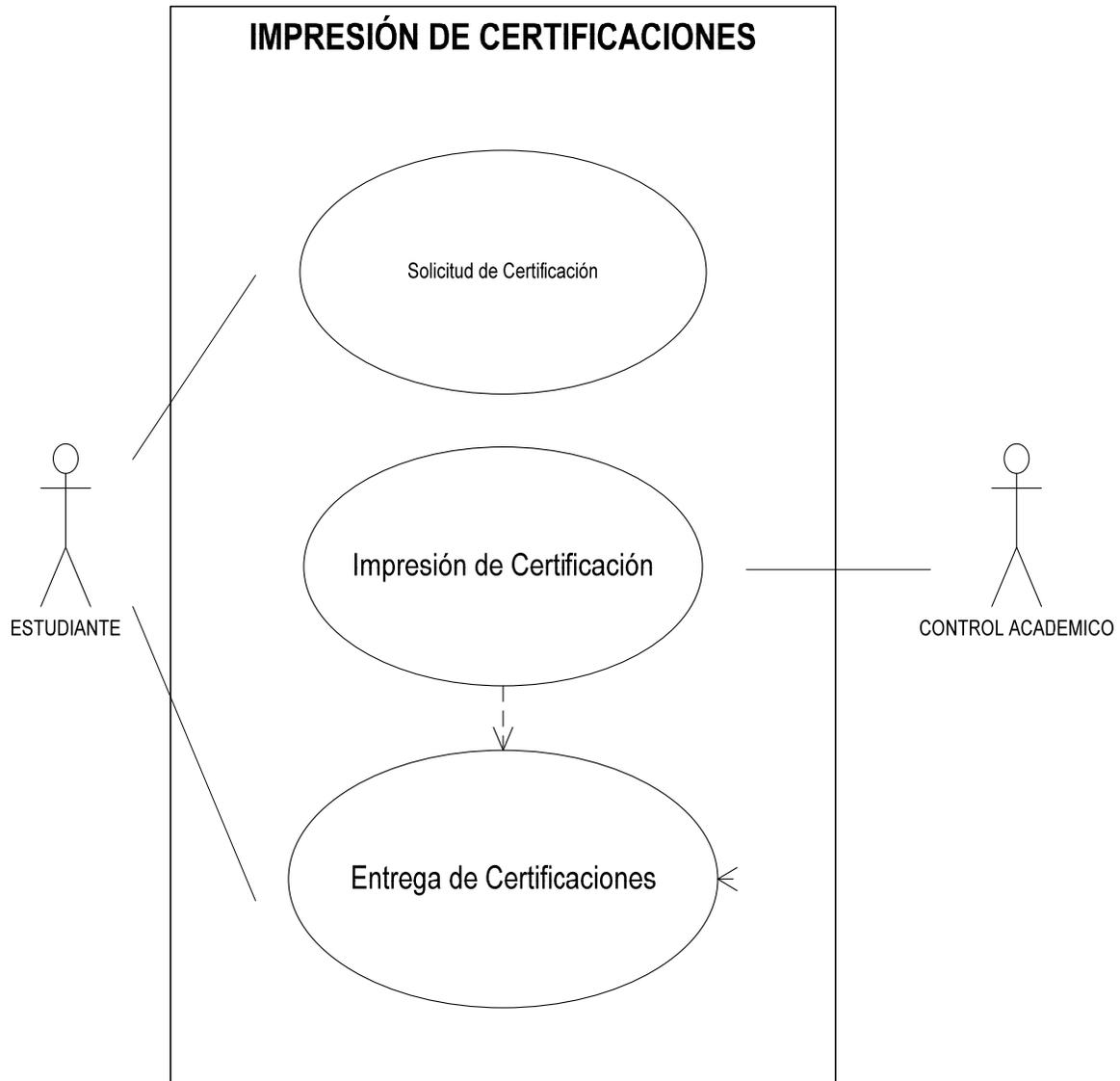
Descripción:

1. El estudiante hace la solicitud de la certificación de cursos.
2. Control Académico imprime las certificaciones de cursos.
3. Estudiante recoge las certificaciones.

Pre-condiciones: 1. Haber ganado cursos asignados.

Frecuencia de Uso: Las veces que se requieran certificaciones de cursos por los estudiantes.

Figura 13. Impresión de certificación de cursos



6. LICENCIA DEL PROYECTO



Copyleft - Software de Control Académico

Para esta licencia se aplican 2 libertades del software libre que son:

Libertad 1: la libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a tus necesidades.

Libertad 3: la libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.

DERECHOS que brinda la licencia:

4. Usar el *software* sin ninguna limitación.
5. Distribuir cuantas copias posee.
6. Transformarlo de la manera conveniente.

OBLIGACIONES que cumplir con la licencia:

- Asegurar las mismas condiciones de uso para el *software* derivado (Ofrecer documentación completa – Especificaciones, manuales, tutoriales).
- Asegurar que la licencia del *software* derivado no pueda ser revocada.
- Facilitar el código fuente del *software* derivado.
- Reconocimiento: debe reconocer los créditos de la obra de manera especificada por el autor o el licenciador.
- El *software* por ninguna razón se puede vender.

CONCLUSIONES

1. Se desarrolló la propuesta de un sistema que permite llevar el control de estudiantes, catedráticos, notas de cursos en forma fácil para la unidad de Control Académico.
2. Este proyecto es un aporte a la Escuela de Sistemas para futuros proyectos que se desarrollen en cuanto a este tema, ya que se dejan las bases para que se pueda seguir modificando un sistema de Control Académico genérico.
3. Con la licencia propuesta en este proyecto, el estudiante que realice su proyecto de EPS, pueda tener acceso a las ideas básicas y enfocarse a las reglas del negocio de cada Escuela / Facultad.
4. Es posible llevar el control de estudiantes de las diferentes carreras de la Escuela Superior de Artes - ESA, así como asignaciones automáticas para los estudiantes de primer ingreso, asignaciones normales para estudiantes de reingreso.
5. Se tiene la información general, localización de los catedráticos que imparten los cursos en los diferentes horarios programados por la Escuela.
6. Se logró cumplir con los objetivos específicos del proyecto, ya que se tienen procesos de asignación, des-asignación de cursos, generación de actas de notas, control de horarios por ciclos planificados por la escuela, generación de actas de cursos a los estudiantes que lo requieran.

RECOMENDACIONES

1. Las siguientes recomendaciones son con la finalidad de seguir con uno de los objetivos de este proyecto, que sea base para futuros proyectos en la misma línea de sistematizar Control Académico de las diferentes Escuelas / Facultades de la Universidad de San Carlos de Guatemala. Las siguientes son:

- ✓ Respetar los lineamientos del tipo de licencia, con que fue desarrollado este proyecto.
- ✓ Que las nuevas versiones del software, se puedan actualizar/acoplar a la PROPUESTA DE UN SISTEMA INICIAL DE CONTROL ACADEMICO PARA LA ESCUELA SUPERIOR DE ARTES (ESA) DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA, para que sean beneficiados con las nuevas tecnologías.
- ✓ Se puede desarrollar la página para que los estudiantes sean beneficiados con los siguientes servicios:
 - a. Asignación de cursos para los estudiantes de primer ingreso o reingreso.
 - b. Carga de notas.
 - c. Consulta de notas de cursos aprobados.

2. Recomendaciones técnicas del equipo

- a. Debe de existir un lugar adecuado para tener el servidor de datos (Aire acondicionado, instalaciones y cableado / conexiones de red adecuadas, por ejemplo, *wireless*).

REFERENCIAS ELECTRÓNICAS

[2] <http://www.getec.etsit.upm.es/docencia/gproyectos/planificacion/cvida.htm>
(abril 2009)

[3] http://es.wikipedia.org/wiki/Software_libre (marzo 2009)

[4] <http://www.desarrolloweb.com/articulos/efecto-virico-copyleft.html> (febrero 2009)

[5] http://es.wikipedia.org/wiki/Software_no_libre#Software_propietario (abril 2009)

[6] http://es.wikipedia.org/wiki/Diagrama_entidad-relaci%C3%B3n (febrero 2009)

[7] <http://es.wikipedia.org/wiki/Delphi> (marzo 2009)

[8] <http://www.manualcopyleft.net/> (marzo 2009)

BIBLIOGRAFÍA

1. LIMA GONZÁLEZ, SERGIO IVÁN y CORADO CASTELLANOS DE LIMA, ERIKA YESENIA. Desarrollo de un sistema de información electrónico que apoye el Control Académico de la Escuela de Estudios de Postgrado de la Facultad de Ciencias Económicas de la Universidad de San Carlos de Guatemala utilizando software libre. (Tesis noviembre 2006) Guatemala, C.A.
