

UNIVERSIDAD PEDAGÓGICA DE EL SALVADOR

FACULTAD DE CIENCIAS ECONÓMICAS

TÉCNICO EN SISTEMAS DE COMPUTACIÓN



TRABAJO DE GRADUACIÓN

**“CREACIÓN DE UN SISTEMA DE REGISTRO Y CONTROL DE NOTAS,
COMPLEJO EDUCATIVO CATÓLICO “SAN JOSÉ”, QUEZALTEPEQUE, LA
LIBERTAD, 2009”**

DOCENTE ORIENTADOR: ING. MELVIN RAÚL CAMPOS.

INTEGRANTES:

OSCAR ERNESTO GUILLÉN GARCÍA.

SANDRA PATRICIA RAMÍREZ BERDUGO.

WILLIAM ALFREDO BELTRÁN NAJARRO.

SAN SALVADOR, OCTUBRE DE 2009

ÍNDICE

CAPITULO I

DATOS DE IDENTIFICACIÓN	Pág. 4
1.1 TEMA	Pág. 5
1.2 OBJETIVOS	Pág. 6
1.3 INTRODUCCIÓN	Pág. 7
1.4 PLANTEAMIENTO DEL PROBLEMA	Pág. 9
1.5 JUSTIFICACIÓN	Pág. 11
1.6 FUNDAMENTACIÓN TEÓRICA	Pág. 13
1.6.1 PROGRAMA	Pág. 13
1.6.2 LENGUAJE DE PROGRAMACIÓN	Pág. 14
1.6.3 BASE DE DATOS	Pág. 16
1.6.4 VISUAL BASIC	Pág. 19
1.6.5 SQL	Pág. 22
1.6.6 EL MODELO ENTIDAD RELACIÓN	Pág. 23
1.6.7 EL LENGUAJE UML	Pág. 26
1.6.8 NORMALIZACIÓN	Pág. 29
1.6.9 SISTEMA	Pág. 30
1.6.10 CICLO DE VIDA DE LOS SISTEMAS	Pág. 31

CAPITULO II

DATOS DE IDENTIFICACIÓN	Pág. 33
2.1 METODOLOGÍA DESCRIPTIVA	Pág. 34
2.1.1 METODOLOGÍA DE LA INVESTIGACIÓN	Pág. 34
2.1.1.1 TIPO DE ESTUDIO	Pág. 34
2.1.1.1.1 ESTUDIO CORRELACIONAL	Pág. 34
2.1.1.1.2 DOCUMENTAL O BIBLIOGRÁFICA	Pág. 34
2.1.1.1.3 DESCRIPTIVA	Pág. 34
2.1.1.1.5 DE CAMPO	Pág. 35
2.2 PROCEDIMIENTOS PARA LA OBTENCIÓN DE DATOS	Pág. 35
2.2.1 TÉCNICAS	Pág. 35

2.2.2 INSTRUMENTO	Pág. 35
2.2.3.2 LA ENTREVISTA	Pág. 38
2.2.3.3 LA OBSERVACIÓN	Pág. 38
2.2.3.4 RECOLECCIÓN DE DATOS	Pág. 38
CAPITULO III	
MARCO OPERATIVO	Pág. 39
3.1 ANÁLISIS DEL DISEÑO	Pág. 40
3.1.1 IDENTIFICACIÓN Y DECISIÓN DE REQUERIMIENTOS	Pág. 40
3.1.2 DIAGRAMAS DE CASOS DE USO	Pág. 44
3.1.3 DESCRIPCIÓN DE CASOS DE USO	Pág. 47
3.1.4 DIAGRAMA ENTIDAD RELACIÓN	Pág. 53
3.1.5 MAPEO	Pág. 54
3.1.6 MODELO CONCEPTUAL	Pág. 56
3.1.7 MODELO FÍSICO	Pág. 57
3.1.8 DIAGRAMA DE CLASES	Pág. 58
3.1.9 DICCIONARIO DE DATOS DEL SISTEMA	Pág. 59
3.2 NUEVAS FORMULACIONES TEÓRICAS	Pág. 62
3.2.1 BENEFICIOS Y VENTAJAS DE LA PROPUESTA	Pág. 62
CONCLUSIONES	Pág. 63
BIBLIOGRAFÍA	Pág. 64
ANEXOS	Pág. 65

CAPITULO I

DATOS DE IDENTIFICACIÓN

1.1 TEMA:

“CREACIÓN DE UN SISTEMA DE REGISTRO Y CONTROL DE NOTAS, COMPLEJO EDUCATIVO CATÓLICO “SAN JOSÉ”, QUEZALTEPEQUE, LA LIBERTAD, 2009”

1.2 OBJETIVOS

OBJETIVO GENERAL

- Desarrollar un sistema para el registro y control de notas académicas que simplifique y agilice de manera eficiente este proceso en el Complejo Educativo Católico “San José” ,Municipio de Quezaltepeque, Departamento de La Libertad.

OBJETIVOS ESPECÍFICOS

- Evaluar el sistema informático instalado para registro y control de Notas del Complejo Educativo Católico “San José”, Municipio de Quezaltepeque, Departamento de La Libertad, para verificar que cumpla con las expectativas propuestas.
- Codificar los programas que permitan agilizar los procesos y actividades del complejo educativo

1.3 INTRODUCCIÓN

La sociedad actual se ha visto forzada, gracias a la tecnología a actualizarse gradualmente para no verse en desventaja ante un entorno tan cambiante. Las nuevas técnicas y procedimientos que se están aplicando tanto en las empresas del sector privado o público, centro escolares, etc. Promueven que éstas agilicen sus actividades para volverse más competitivas y eficientes.

Los últimos cincuenta años han sido testigos del vertiginoso desarrollo tecnológico en donde los bits están más al alcance de todo el mundo en la mayoría de áreas del quehacer cotidiano.

La finalidad de la investigación es conocer la aplicación práctica que tiene el software en un proceso administrativo, se ha decidido trabajar en el sector público, específicamente en un Centro Escolar que, pese a gozar del equipo tecnológico necesario, la mayoría de sus procesos se realizan en forma manual. Luego de visitar la institución y analizar las diferentes opciones problemáticas en las cuales se podría trabajar, se decidió encausar la investigación al área que comúnmente ocupa mayor esfuerzo y tiempo, especialmente cuando se trata de obtener resultados rápidos y oportunos, el registro y control de notas.

Para su comprensión el documento se estructura bajo el siguiente capitulado:

Capítulo I, Planteamiento del problema, el cual se inicia con la situación problemática donde se hace una descripción del problema en estudio, de dicha situación se enuncia el problema que guió el desarrollo de la investigación. En la justificación, se plantean las causas por las que se seleccionó el tema a investigar, la importancia del estudio, la utilidad del mismo y sus beneficios a la comunidad, se presentan los objetivos: uno general y dos específicos, los cuales dan los propósitos a cumplir con la presente investigación, luego también se incluirán la fundamentación teórica para dar sustento a nuestra investigación.

Capitulo II, en este capítulo se plantea el tipo de estudio, la población y muestra, los métodos, técnicas e instrumentos y los procedimientos correspondientes.

Capitulo III, se hará referencia al documento de análisis del sistema lo cual nos ayudara a entender de que podrá hacer el programa a la hora de ejecución.

Como documentos complementarios se presenta la bibliografía consultada y los anexos correspondientes.

1.4 PLANTEAMIENTO DEL PROBLEMA

El Salvador como país perteneciente al continente Centro Americano no escapa del alto crecimiento poblacional en general y por lo tanto un alto crecimiento de la población apto para prepararse académicamente en el nivel de educación, sin dejar de lado diferentes circunstancias que interfieren en la formación académica de cada uno de sus habitantes (pobreza, desempleo, dificultad para acceder al sistema educativo, violencia y justicia social, etc.); sin embargo el gobierno de El Salvador hace grandes esfuerzos para brindar a la población educación pública que le permita cubrir la demanda educativa a nivel básico, medio y superior pues cada día las exigencias por erradicar el índice de analfabetismo cobran mayor protagonismo a tal grado que organizaciones mundiales que velan por los intereses de las personas con menos recursos en todos sus aspectos, intervienen aportando ayuda económica a los países mas pobres del mundo, para contribuir a una mejor formación personal, cultural y educativa de la población, a través de programas orientados al desarrollo de políticas educativas, fiscales, de salud y desarrollo social para que estas permitan impulsar la educación básica, media y superior a los lugares mas remotos de El Salvador. Por lo tanto es importante enfocar la inversión desde la perspectiva educacional; así como lineamientos del estado que permitan no solo aperturar centros educativos y brindar la educación, si no buscar que la educación se convierta en una fuente de desarrollo para la población Salvadoreña. El Ministerio de Educación, como ente rector de la preparación académica de los habitantes de El Salvador, con el afán de satisfacer

la demanda educativa de la población en general, cuenta con centros educativos de nivel básico, media y superior. Una de estas instituciones es el Complejo Educativo Católico “San José” del Municipio de Quezaltepeque Departamento de La Libertad, el cuál fue fundado en el año de 1920, con la finalidad de brindar cobertura educativa a la población del área urbana y rural de dicho municipio con mas de 1,000 alumnos. Esta actividad se complementa con el registro y control académico el cual contiene la información relacionada a la formación educativa de sus estudiantes. Actualmente el COMPLEJO EDUCATIVO CATÓLICO “SAN JOSÉ” no cuenta con un sistema informático que permita llevar un control de información y registro académico de los alumnos; pues toda la información es llevada de forma manual solamente en formularios donde se detalla el rendimiento académico de cada alumno, situación que esta originando problemas como: el tiempo para actualizar la información (calcular porcentajes para laboratorios, exámenes y promediar notas) para luego realizar, analizar y proporcionar los reportes requeridos. Agilizar y proporcionar un control, eficaz y eficiente, del registro académico de cada uno de los alumnos, así como tener información de forma actualizada se vuelven indispensables y necesarios, pues debido a la acumulación y difícil control del registro académico de cada uno de los alumnos, es necesario optimizar dicha situación mediante un sistema informático como una alternativa para solucionar la dificultad de llevar un ágil control y registro académico de los alumnos.

1.5 JUSTIFICACIÓN

Los sistemas de información mediante la computadora, en la actualidad, constituyen una necesidad esencial hoy en día que cualquier institución pública o privada cuente con sistemas automatizados para llevar el control de las diferentes actividades que estas deben cubrir de acuerdo a su naturaleza y sus funciones, es necesario llevar a cabo una investigación para diseñar un sistema de registro y control de notas académicas en el Complejo Educativo Católico “San José” , del Municipio de Quezaltepeque, Departamento de La Libertad, para realizar la automatización de dicho registro a través de un sistema informático, el cual permita a este centro educativo manejar un sistema de información de todos los alumnos (de primero a noveno grado y bachillerato) modernizando y mejorando el registro académico de una manera adecuada para cubrir una de las necesidades que en la actualidad enfrenta la institución, debido a la cantidad de población estudiantil con que cuenta.

Con el diseño del sistema informático se pretende contribuir a brindar un mejor Servicio de notas académicas de todos los Alumnos que estudian en el Complejo Educativo Católico “San José”, distribuidos en los turnos matutinos y vespertinos; los cuales se verán beneficiados con el diseño de este sistema, que contribuirá a minimizar, mejorar y agilizar por medio del programa visual basic porque posee una curva de aprendizaje muy rápida, integra el diseño e implementación de formularios de Windows, permite usar con suma facilidad la

plataforma de los sistemas, el código en Visual Basic es fácilmente migrable a otros lenguajes y para realizar la base de datos se usara el programa SQL este gestor de bases de datos se utilizara, debido a su gran rapidez y facilidad de uso.

Con el desarrollo e implementación de éste sistema se obtendrá:

- 1- El Registro y actualización de notas y promedios.
- 2- La seguridad en el manejo de la información.
- 3- Los reportes de notas académicas requeridos por padres de familia y demás reportes necesarios para el centro escolar.
- 4- Tiempo para la entrega de notas por períodos y al final de cada año lectivo a los padres de familia.

1.6 FUNDAMENTACIÓN TEÓRICA

1.6.1 PROGRAMA

Un programa es una serie de comandos ejecutados por el equipo. Sin embargo, el equipo sólo es capaz de procesar elementos binarios, es decir, una serie de 0s y 1s. Por lo tanto, necesitamos un lenguaje de programación para escribir de manera legible, es decir, con comandos que el ser humano pueda comprender (por ser similares a su propio lenguaje) los comandos que el equipo deberá ejecutar.

El método de escritura de un programa está muy ligado al lenguaje de programación elegido, ya que existen muchos tipos distintos. Además, el compilador debe coincidir con el lenguaje elegido: cada lenguaje de programación tiene su propio compilador (excepto los lenguajes interpretados).

De acuerdo a sus funciones, los programas pueden ser clasificados en software de sistema y software de aplicación. Pueden ejecutarse muchos programas de forma simultánea en un mismo ordenador, a lo cual se le llama multitarea.

1.6.2 LENGUAJE DE PROGRAMACIÓN

Con la aparición de las computadoras desaparecen las secuencias de posiciones de llaves mecánicas que debían desconectarse para obtener una acción determinada, una llave conectada era un 1 y una llave desconectada era un 0. Una sucesión de llaves en cualquiera de sus dos posiciones definía una secuencia de ceros y unos (por ejemplo: 0100011010011101...) que venía a representar una instrucción o un conjunto de instrucciones (programa) para el ordenador (o computador) en el que se estaba trabajando. A esta primera forma de especificar programas para una computadora se la denomina lenguaje máquina o código máquina.

La necesidad de recordar secuencias de programación para las acciones usuales llevó a denominarlas con nombres fáciles de memorizar y asociar: ADD (sumar), SUB (restar), MUL (multiplicar), CALL (ejecutar subrutina), etc. A esta secuencia de posiciones se le denominó "instrucciones", y a este conjunto de instrucciones se le llamó lenguaje ensamblador.

Posteriormente aparecieron diferentes lenguajes de programación, los cuales reciben su denominación porque tienen una estructura sintáctica similar a los lenguajes escritos por los humanos.

Los lenguajes de programación sirven para escribir programas que permite la comunicación usuario y maquina. Unos programas especiales llamados traductores (compiladores o intérpretes) convierten las instrucciones inscritas en

lenguajes de programación en instrucciones escritas en lenguajes de maquina (0 y 1, bits) que esta puede entender.

Los principales tipos de lenguaje utilizados en la actualidad son:

- Lenguaje de máquina
- Lenguaje de bajo nivel (ensamblador)
- Lenguaje de alto nivel

1.6.3 BASE DE DATOS

Base de Datos es un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en máquina accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo.

Ventajas del uso de la base de datos en la organización

Ventajas de las bases de datos.

1. Independencia de datos y tratamiento.

Cambio en datos no implica cambio en programas y viceversa (Menor coste de mantenimiento).

2. Coherencia de resultados.

Reduce redundancia:

- Acciones lógicamente únicas.
- Se evita inconsistencia.

3. Mejora en la disponibilidad de datos

No hay dueño de datos (No igual a ser públicos).

- Ni aplicaciones ni usuarios.
- Guardamos descripción (Idea de catálogos).

4. Cumplimiento de ciertas normas.

Restricciones de seguridad.

- Accesos (Usuarios a datos).
- Operaciones (Operaciones sobre datos)

5. Otras ventajas:

- Más eficiente gestión de almacenamiento
- Efecto sinérgico.

Los SBD pueden ser desde 3 niveles distintos:

1. Nivel Físico.

Es el nivel real de los datos almacenados. Es decir como se almacenan los datos, ya sea en registros, o como sea. Este nivel es usado por muy pocas personas que deben estar cualificadas para ello. Este nivel lleva asociada una representación de los datos, que es lo que denominamos Esquema Físico.

2.- Nivel Conceptual.

Es el correspondiente a una visión de la base de datos desde el punto de vista del mundo real. Es decir tratamos con la entidad u objeto representado, sin importarnos como está representado o almacenado. Este nivel lleva asociado el Esquema Conceptual.

3.- Nivel Visión.

Son partes del esquema conceptual. El nivel conceptual presenta toda la base de datos, mientras que los usuarios por lo general sólo tienen acceso a pequeñas parcelas de ésta. El nivel visión es el encargado de dividir estas parcelas.

1.6.4 VISUAL BASIC. NET

Es uno de los tantos lenguajes de programación que podemos encontrar hoy en día. Dicho lenguaje nace del BASIC (Beginner's All-purpose Symbolic Instruction Code) que fue creado en su versión original en el Dartmouth College, con el propósito de servir a aquellas personas que estaban interesadas en iniciarse en algún lenguaje de programación. Luego de sufrir varias modificaciones, en el año 1978 se estableció el BASIC estándar. La sencillez del lenguaje ganó el desprecio de los programadores avanzados por considerarlo "un lenguaje para principiantes".

Primero fue GW-BASIC, luego se transformó en QuickBasic y actualmente se lo conoce como Visual Basic y la versión más reciente es la 6 que se incluye en el paquete Visual Studio 6 de Microsoft. Esta versión combina la sencillez del BASIC con un poderoso lenguaje de programación Visual que juntos permiten desarrollar robustos programas de 32 bits para Windows. Esta fusión de sencillez y la estética permitió ampliar mucho más el monopolio de Microsoft, ya que el lenguaje sólo es compatible con Windows, un sistema operativo de la misma empresa.

Visual Basic ya no es más "un lenguaje para principiantes" sino que es una perfecta alternativa para los programadores de cualquier nivel que deseen desarrollar aplicaciones compatibles con Windows.

En este informe explicaremos algunos términos y/o características de mismo con la finalidad de aprender más sobre este Programa y manejarlo con facilidad

PARTES DEL ENTORNO DE VISUAL BASIC.

- Barra de título: muestra el nombre del proyecto y del formulario q se está diseñando actualmente
- Barra de menús: agrupa los menús despegables que contienen todas las operaciones que pueden llevarse a cabo con Visual Basic 6.0.
- Barra de herramientas estándar: contienen los botones que se utilizan con mayor frecuencia cuando se trabaja con un proyecto. Simplifica la elección de opciones de los menús Archivo, Edición, Ver y Ejecutar; además, en el área derecha presenta la ubicación (coordenadas) y el tamaño del objeto seleccionado
- Ventana de formulario: es el área donde se diseña la interfaz gráfica, es decir, es donde se inserta electo gráficos, como botones, imágenes, casilla de verificación, cuadros de listas, etc.
- Cuadro de herramientas: presenta todos los controles necesarios para diseñar una aplicación, como cuadros de texto, etiquetas, cuadros de listas, botones de comandos, etc.
- Ventana de proyecto: muestra los elementos involucrados en el proyecto, como formularios, módulos, controles oxc, etc. cada elemento puede seleccionarse en forma independiente para su edición.

- Ventana de posición del formulario: muestra la ubicación que tendrá el formulario en la pantalla, cuando ejecute la aplicación. Esta ubicación puede cambiarse si se hace clic con el botón izquierdo del mouse.
- La Ventana propiedades muestra todas las propiedades del control actualmente seleccionado, en este caso muestra las propiedades del Form1, luego podemos ver que abajo dice "Form1 Form", lo que está en negrita es el nombre del objeto, y lo que le sigue es el tipo de objeto, en este caso es un Formulario (Form).

1.6.5 SQL

Es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, acceder ha y procesar datos guardados en un computador, un administrador como SQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

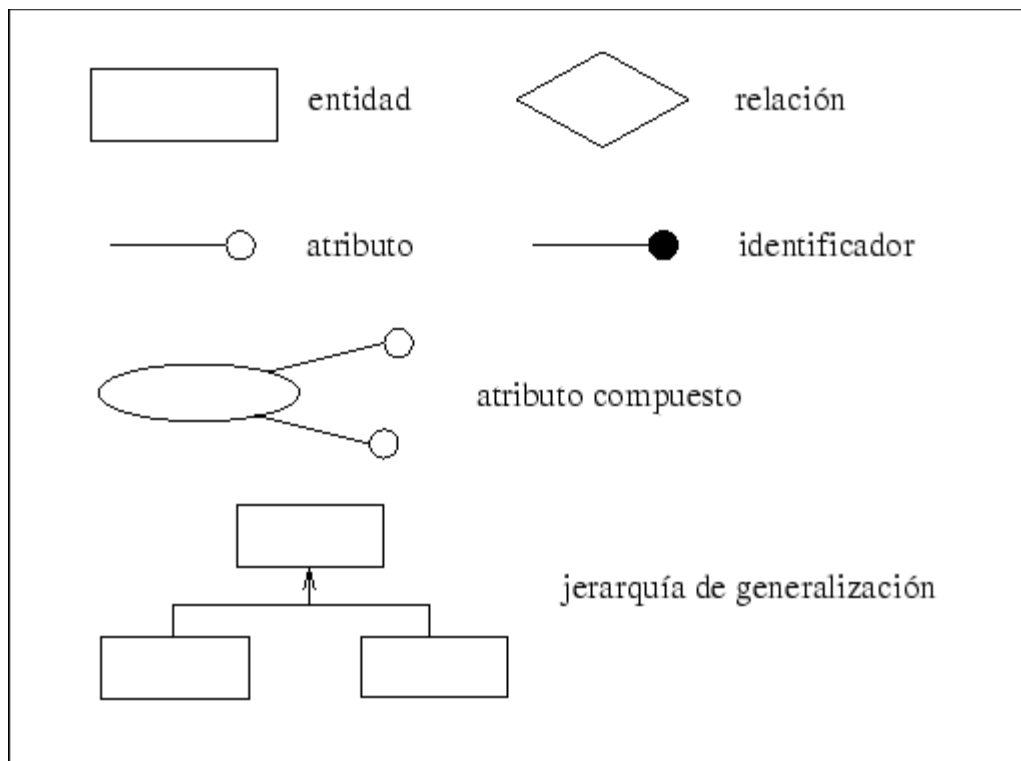
Las características generales de este gestor de bases de datos son las siguientes:

1. El SQL es un lenguaje de acceso a bases de datos que explota la flexibilidad y potencia de los sistemas relacionales permitiendo gran variedad de operaciones en éstos últimos.
2. Es un lenguaje declarativo de "alto nivel" o "de no procedimiento", que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación y la orientación a objetos. De esta forma una sola sentencia puede equivaler a uno o más programas que utilizas en un lenguaje de bajo nivel orientado a registro.

1.6.6 EL MODELO ENTIDAD- RELACIÓN

El modelo entidad-relación es el modelo conceptual más utilizado para el diseño conceptual de bases de datos. Fue introducido por Peter Chen en 1976. El modelo entidad-relación está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

Originalmente, el modelo entidad-relación sólo incluía los conceptos de entidad, relación y atributo. Más tarde, se añadieron otros conceptos, como los atributos compuestos y las jerarquías de generalización, en lo que se ha denominado modelo entidad-relación extendido.



Entidad

Cualquier tipo de objeto o concepto sobre el que se recoge información: cosa, persona, concepto abstracto o suceso. Por ejemplo: coches, casas, empleados, clientes, empresas, oficios, diseños de productos, conciertos, excursiones, etc.

Las entidades se representan gráficamente mediante rectángulos y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Relación (interrelación)

Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su función. Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.

Las entidades que están involucradas en una determinada relación se denominan entidades participantes.

Atributo

Es una característica de interés o un hecho sobre una entidad o sobre una relación. Los atributos representan las propiedades básicas de las entidades y de las relaciones. Toda la información extensiva es portada por los atributos. Gráficamente, se representan mediante bolitas que cuelgan de las entidades o relaciones a las que pertenecen.

Cada atributo tiene un conjunto de valores asociados denominado dominio. El dominio define todos los valores posibles que puede tomar un atributo. Puede haber varios atributos definidos sobre un mismo dominio.

1.6.7 EL LENGUAJE UML

UML= (Unified Modeling Language) Lenguaje Unificado de Moldeamiento

Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema. Un sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que finalmente beneficia a un usuario externo.

El lenguaje de modelado pretende unificar la experiencia pasada sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.

Objetivos del UML

- UML es un lenguaje de modelado de propósito general que pueden usar todos los modeladores. No tiene propietario y está basado en el común acuerdo de gran parte de la comunidad informática.
- UML no pretende ser un método de desarrollo completo. No incluye un proceso de desarrollo paso a paso. UML incluye todos los conceptos que se consideran necesarios para utilizar un proceso moderno, basado en construir una sólida arquitectura para resolver requisitos dirigidos por casos de uso.

- Ser tan simple como sea posible pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir. UML necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, como son la encapsulación y componentes.
- Debe ser un lenguaje universal, como cualquier lenguaje de propósito general.
- Imponer un estándar mundial.

Arquitectura del UML

Arquitectura de cuatro capas, definida a fin de cumplir con la especificación Meta Object Facility del OMG:

- Meta-metamodelo: define el lenguaje para especificar metamodelos.
- Metamodelo: define el lenguaje para especificar modelos.
- Modelo: define el lenguaje para describir un dominio de información.
- Objetos de usuario: define un dominio de información específico.

Áreas conceptuales de UML

Los conceptos y modelos de UML pueden agruparse en las siguientes áreas conceptuales:

Estructura estática:

Cualquier modelo preciso debe primero definir su universo, esto es, los conceptos clave de la aplicación, sus propiedades internas, y las relaciones entre cada una de ellas. Este conjunto de construcciones es la estructura estática.

Comportamiento dinámico:

Hay dos formas de modelar el comportamiento, una es la historia de la vida de un objeto y la forma como interactúa con el resto del mundo y la otra es por los patrones de comunicación de un conjunto de objetos conectados, es decir la forma en que interactúan entre sí.

Construcciones de implementación:

Los modelos UML tienen significado para el análisis lógico y para la implementación física. Un componente es una parte física reemplazable de un sistema y es capaz de responder a las peticiones descritas por un conjunto de interfaces.

Organización del modelo:

La información del modelo debe ser dividida en piezas coherentes, para que los equipos puedan trabajar en las diferentes partes de forma concurrente. El conocimiento humano requiere que se organice el contenido del modelo en paquetes de tamaño modesto.

1.6.8 NORMALIZACIÓN

Normalización es un proceso que clasifica relaciones, objetos, formas de relación y demás elementos en grupos, en base a las características que cada uno posee. Si se identifican ciertas reglas, se aplica una categoría; si se definen otras reglas, se aplicará otra categoría.

Estamos interesados en particular en la clasificación de las relaciones BDR. La forma de efectuar esto es a través de los tipos de dependencias que podemos determinar dentro de la relación. Cuando las reglas de clasificación sean más y más restrictivas, diremos que la relación está en una forma normal más elevada. La relación que está en la forma normal más elevada posible es que mejor se adapta a nuestras necesidades debido a que optimiza las condiciones que son de importancia para nosotros:

- La cantidad de espacio requerido para almacenar los datos es la menor posible;
- La facilidad para actualizar la relación es la mayor posible;
- La explicación de la base de datos es la más sencilla posible.

1.6.9 SISTEMA

Es el conjunto de partes interrelacionadas, hardware, software y de Recurso Humano. Un sistema informático típico emplea una computadora que usa dispositivos programables para capturar, almacenar y procesar datos. La computadora personal o PC, junto con la persona que lo maneja y los periféricos que los envuelven, resultan de por sí un ejemplo de un sistema informático. Incluso la computadora más sencilla se clasifica como un sistema informático, porque al menos dos componentes (hardware y software) tienen que trabajar unidos. Pero el genuino significado de "sistema informático" viene mediante la interconexión. Muchos sistemas informáticos pueden interconectarse, esto es, unirse para convertirse un sistema mayor. La interconexión de sistemas informáticos puede tornarse difícil debido a incompatibilidades. A veces estas dificultades ocurren a nivel de hardware, mientras que en otras ocasiones se dan entre programas informáticos que no son compatibles entre sí.

Los diseñadores de sistemas informáticos no necesariamente esperan que sus sistemas se puedan interconectar con otros sistemas. Por otro lado, los técnicamente eruditos a menudo pueden configurar sistemas diferentes para que se puedan comunicar entre sí usando un conjunto de reglas y restricciones conocidas como protocolos. Los protocolos tratan precisamente de definir la comunicación dentro de y entre sistemas informáticos distintos pero conectados entre sí. Si dos sistemas informáticos usan el mismo protocolo, entonces podrán ser capaces de interconectarse y formar parte de un sistema mayor.

1.6.10 CICLO DE VIDA DE LOS SISTEMAS

El término ciclo de vida del software describe el desarrollo de software, desde la fase inicial hasta la fase final. El propósito de este programa es definir las distintas fases intermedias que se requieren para validar el desarrollo de la aplicación, es decir, para garantizar que el software cumpla los requisitos para la aplicación y verificación de los procedimientos de desarrollo: se asegura de que los métodos utilizados son apropiados.

Estos programas se originan en el hecho de que es muy costoso rectificar los errores que se detectan tarde dentro de la fase de implementación. El ciclo de vida permite que los errores se detecten lo antes posible y por lo tanto, permite a los desarrolladores concentrarse en la calidad del software, en los plazos de implementación y en los costos asociados.

El ciclo de vida básico de un software consta de los siguientes procedimientos:

- Definición de objetivos: definir el resultado del proyecto y su papel en la estrategia global.
- Análisis de los requisitos y su viabilidad: recopilar, examinar y formular los requisitos del cliente y examinar cualquier restricción que se pueda aplicar.
- Diseño general: requisitos generales de la arquitectura de la aplicación.
- Diseño en detalle: definición precisa de cada subconjunto de la aplicación.

- Programación (programación e implementación): es la implementación de un lenguaje de programación para crear las funciones definidas durante la etapa de diseño.
- Prueba de unidad: prueba individual de cada subconjunto de la aplicación para garantizar que se implementaron de acuerdo con las especificaciones.
- Integración: para garantizar que los diferentes módulos se integren con la aplicación. Éste es el propósito de la prueba de integración que está cuidadosamente documentada.
- Prueba beta (o validación), para garantizar que el software cumple con las especificaciones originales.
- Documentación: sirve para documentar información necesaria para los usuarios del software y para desarrollos futuros.
- Mantenimiento: para todos los procedimientos correctivos (mantenimiento correctivo) y las actualizaciones secundarias del software (mantenimiento continuo).

El orden y la presencia de cada uno de estos procedimientos en el ciclo de vida de una aplicación dependen del tipo de modelo de ciclo de vida acordado entre el cliente y el equipo

CAPITULO II

MARCO EMPÍRICO

2.1 METODOLOGÍA DESCRIPTIVA.

2.1.1 Metodología de la Investigación

2.1.1.1 Tipo de estudio:

La investigación se hizo de tipo correlacional porque se aplicaron procedimientos documentales y bibliográficos, y un análisis descriptivo para efectos de procesos de los datos Obtenidos en la fase de campo.

2.1.1.1.1 Estudio correlacional: Se Estableció el grado de correlación existente entre las variables independientes y la dependiente y los resultados derivados de la relación entre dichas variables.

2.1.1.1.2 Documental o bibliográfica: Nos permitió consultar una serie de fuentes: libros, folletos, tesis y otro tipo de bibliografía que contribuya a sustentar la parte teórica del documento.

2.1.1.1.3 Descriptiva: se describieron indicadores que componen las variables del nivel para relacionarlo con el sistema de Registro y control de Notas en el Complejo Educativo “San José” del municipio de Quezaltepeque del departamento de la libertad.

2.1.1.1.5 De campo: porque nos desplazamos al municipio de Quezaltepeque donde realizamos una entrevista para recolectar información.

2.2 PROCEDIMIENTOS PARA LA OBTENCIÓN DE DATOS

2.2.1 Técnicas

Las técnicas que incluimos para el desarrollar la investigación fue: la entrevista y la observación. La entrevista se realizo de manera directa del investigador y el sujeto de estudio se obtuvo respuestas verbales a las interrogantes realizadas sobre el tema de investigación, la observación directa permitió registrar información visual de características de los sujetos de la investigación, relacionada con las variables del estudio.

2.2.2 Instrumento

Para la realización de esta investigación se utilizó una entrevista estructurada conformada por 10 preguntas, diseñadas de acuerdo a los indicadores en la operacionalización de hipótesis dentro del sistema.

OBJETIVO:

- **Conocer las necesidades del sistema y sus errores**

A manera de introducción la entrevista con “x” persona puede iniciarse así:

- ✚ Primero saludamos
- ✚ Nos presentamos quienes somos y cuales son nuestras intensiones
- ✚ Preguntamos el nombre de el (a).

RESULTADO DE LA ENTREVISTA REALIZADA AL ADMINISTRADOR DE CONTROL DE NOTAS DE LOS ALUMNOS DEL COMPLEJO EDUCATIVO “SAN JOSÉ”

PREGUNTAS:

1. ¿Cuántas personas llevan el registro y el control de notas?

R// Solo una persona que es el encargado de informática, los maestros digitan las notas y se las dan al administrador.

2. ¿Cuál es su opinión del estado actual del registro y e control de notas?

R// Tiene varias debilidades.

No se puede ver el promedio, no se puede ver el proceso por ejemplo: Materia, primer periodo, segundo periodo, tercer periodo.

No cumple con seguridad.

Puede confundir a un profesor al momento de hacer cálculos.

Los porcentajes pueden variar.

3. ¿Qué ventajas y desventajas trae trabajar con ese software?

R//Ventajas.

Excel es muy amigable fácil de utilizar.

El sistema es obsoleto y no funcional.

No responde no se puede hacer cambios.

4. ¿Cuántas veces al año actualizan las notas?

R//Las notas se actualizan cinco veces al año el ultimo es el promedio.

5. ¿Cuánto se tardan en procesar los datos de las notas?

Un día, los profesores entregan una plantilla con las notas al encargado de informática las notas y el se encarga de agregarlas al sistema.

6. ¿Alguna vez se les han perdido archivos?

R//.Esto ya ha pasado con los maestros por que los archivos los tienen almacenados en disquete y al momento de dárselos al encargado de informática no se pueden abrir los archivos, pero ya estando en el sistema no se pierden.

7. ¿Con que frecuencia se les han perdido los archivos?

Casi Nunca

8. ¿Con que facilidad dan mantenimiento al programa actual para ingresar notas?

Excel es muy fácil pero no rápido.

9. ¿Cuál es su opinión en cuanto a la seguridad de los archivos?

No tiene ninguna seguridad.

El encargado de informática le da a cada maestro una plantilla de Excel para que ingresen las notas.

10.¿Le beneficiaría que se realizara un sistema que va mas a la vanguardia de la tecnología?

A cualquiera le gustaría tener un sistema adecuado a las necesidades.

2.2.3.2 La entrevista:

Las entrevistas se utilizaron para obtener información en forma verbal, a través de preguntas que propusimos los analistas.

La estructura de la entrevista varía. Nuestro objetivo era adquirir información del sistema actual y los errores que posee.

2.2.3.3 La Observación:

Otra técnica útil que utilizamos para el proceso de la investigación, consistió en observar a las personas cuando efectúan su trabajo. Como técnica de investigación, la observación tiene amplia aceptación. Nos permite determinar que se está haciendo, como se está haciendo, quien lo hace, cuando se lleva a cabo, cuanto tiempo toma, dónde se hace y por que se hace.

2.2.3.4 Recolección de Datos

Para la recolección de datos participó todo el grupo de investigadores el cual a través de la entrevista estructurada se dirigió al administrador del sistema de registro y control de notas de la institución sobre: la utilización de un sistema de información relacionado con la evaluación y registro de calificaciones de los estudiantes del Complejo Educativo “San José”, del municipio de Quezaltepeque, del departamento de La Libertad, durante el periodo de enero a junio del 2009.

CAPITULO III

MARCO OPERATIVO

3.1 ANÁLISIS DEL DISEÑO.

3.1.1 Identificación y decisión de requerimientos

Para poder implementar los módulos del sistema de registro y control de notas, es necesario tener presente los requisitos o necesidades del sistema, conociendo que se tiene un proceso manual de un sistema, debido a que los procesos realizados actualmente no son totalmente eficaces sino que existen procesos manuales y tiene varias debilidades:

No se puede ver el promedio de los alumnos, no se puede ver el proceso por ejemplo: Materia, primer periodo, segundo periodo, tercer periodo.

No cumple con seguridad.

Puede confundir a un profesor al momento de hacer cálculos.

Los porcentajes pueden variar por periodos.

Para el registro y el control de notas una persona es la encargada y es el administrador, los maestros solamente digitan las notas y las entregan al encargado.

Esto requiere que se implementen nuevas y mejores herramientas para poder brindar al encargado mayor eficacia con respecto al ahorro de tiempo y de procesos. Por lo tanto los módulos se llevaran a cabo mediante la metodología se considero mas conveniente, las técnicas que se utilizaran serán el modelado de datos en ER, la documentación basada en el cual contiene todo el desarrollo del proyecto, además de encuestas y entrevistas a los usuarios para ver los

resultados que se han obtenido con el sistema, y de esta forma poder depurar posibles errores en este caso se tomara en cuenta la realización de una versión de prueba del sistema, para posterior revisión.

Entre las herramientas que serán necesarias tenemos:

Requerimiento de la máquina

La máquina debe tener las siguientes características mínimas necesaria para ejecutar dicho programa:

- Que tengan espacio libre de 2 gigabyte
- Que sea Pentium IV
- Que tenga 256 de memoria RAM
- Que tenga Windows XP o superior

Requisitos de Software

En la creación de los módulos que constituirán el sistema se piensan utilizar las siguientes herramientas.

- Sistema Operativo Windows XP
- Microsoft Visual.Net 2005
- Microsoft SQL Server 2005

Recurso Humano

- Encargado de ingreso de datos
- Administrador

Plataforma a trabajar

Sobre que plataforma se va hacer el programa es el visual Basic .Net orientado a objetos

Cotización de licencia



Cotización de licencia de 2005

Compras individuales a establecimientos	Versión Actualización	Completa Versión
Visual Studio 2005 Professional Edition	\$549.00	\$799.00
Visual Studio 2005 Tools for Office	\$549.00	\$799.00
Visual Studio 2005 Standard Edition	\$199.00	\$299.00
Visual Source Safe 2005	\$279.00	\$549.00
Visual Studio 2005 Team Test Load Agent	N/A	\$5,089.00
	CAL	Servidor
Visual Studio 2005 Team Foundation Server	\$499.00	\$2,799.00
Compras individuales con una suscripción a MSDN	Renovación	Nuevo
Visual Studio 2005 Team Suite con suscripción Premium a MSDN	\$3,499.00	\$10,939.00
Visual Studio 2005 Team Edition for Software Architects con suscripción Premium a MSDN	\$2,299.00	\$5,469.00
Visual Studio 2005 Team Edition for Software Developers con suscripción Premium a MSDN	\$2,299.00	\$5,469.00
Visual Studio 2005 Team Edition for Software Testers con suscripción Premium a MSDN	\$2,299.00	\$5,469.00
Visual Studio 2005 Professional Edition con suscripción Premium a MSDN	\$1,999.00	\$2,499.00
Visual Studio 2005 Professional Edition con suscripción Premium a MSDN	\$799.00	\$1,199.00
suscripción a sistemas operativos de Microsoft	\$499.00	\$699.00
Suscripción a la biblioteca de Microsoft	\$99.00	\$199.00

ESPECIFICACIÓN DE REQUISITOS DE USUARIO

Luego de una entrevista con el encargado hemos recopilado los siguientes requisitos.

Requisitos funcionales:

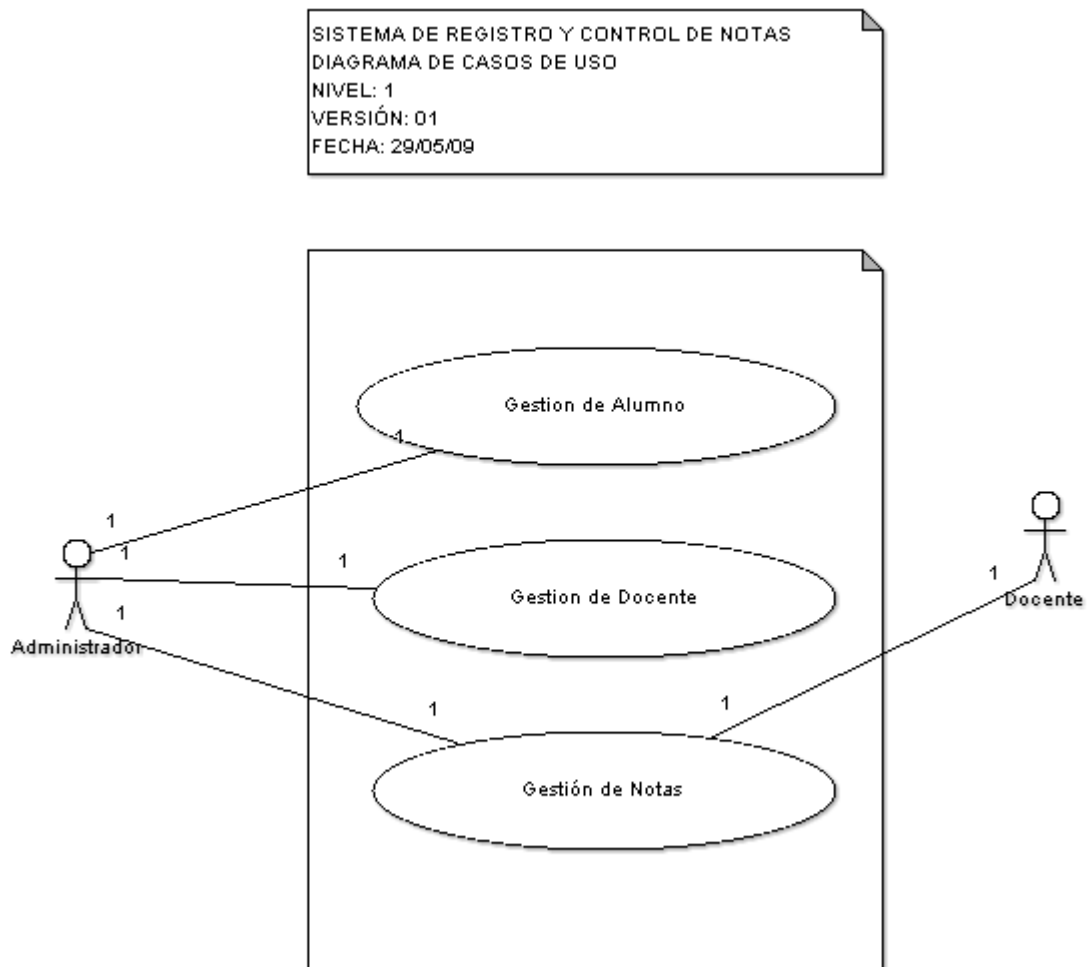
- El sistema deberá almacenar información del alumno.
- El sistema permitirá modificar la información del alumno.
- El sistema permitirá eliminar la información del alumno.
- El sistema deberá almacenar información del docente.
- El sistema permitirá modificar la información del docente.
- El sistema permitirá eliminar la información del docente.
- El sistema deberá almacenar notas.
- El sistema permitirá modificar notas.
- El sistema permitirá eliminar notas.
- El sistema permitirá consultar notas.

Requisitos No Funcionales:

- El usuario debe contar con un computador, que cumpla con las necesidades del sistema.
- El software debe correr sobre Windows.
- El software debe soportar una gran cantidad de datos.
- El usuario del sistema, deberá tener conocimientos básicos de computación.

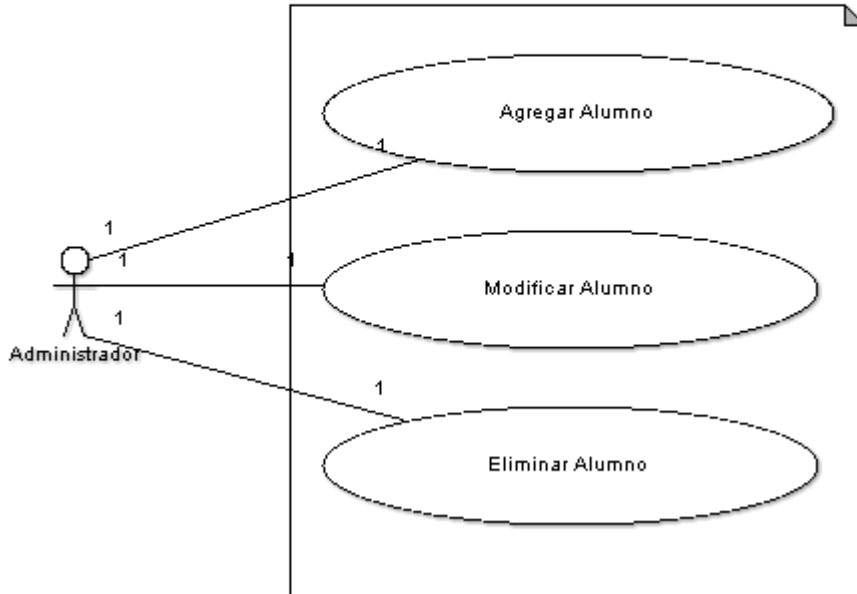
3.1.2 DIAGRAMA DE CASOS DE USO

NIVEL: 1

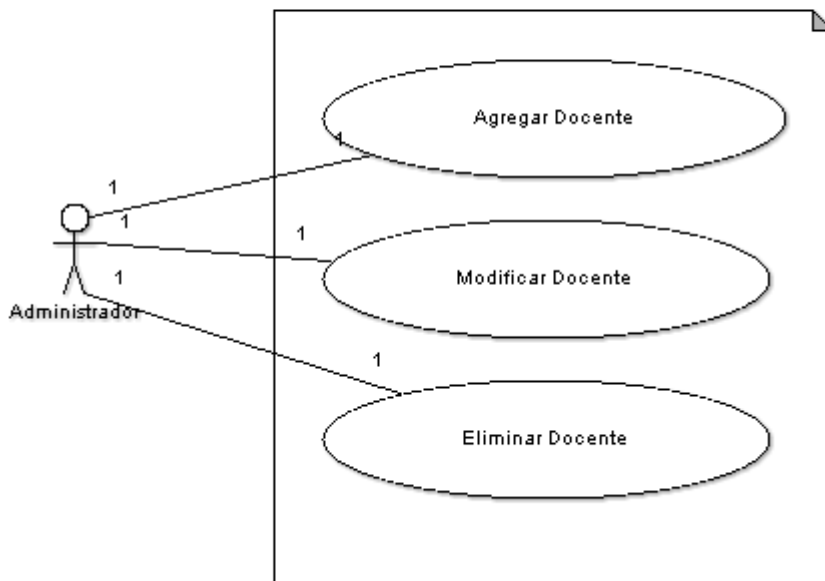


NIVEL: 2

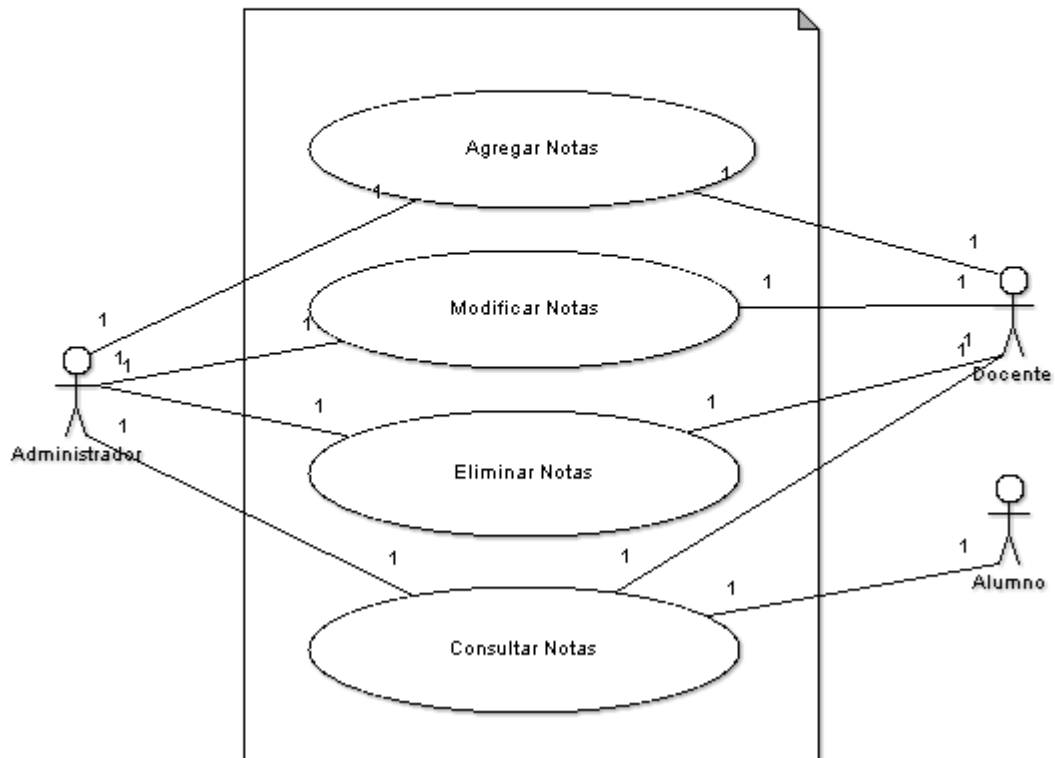
SISTEMA DE REGISTRO Y CONTROL DE NOTAS
DIAGRAMA DE CASOS DE USO
NIVEL: 2 GESTIÓN DE ALUMNO
VERSIÓN: 01
FECHA: 29/05/09



SISTEMA DE REGISTRO Y CONTROL DE NOTAS
DIAGRAMA DE CASOS DE USO
NIVEL: 2 GESTIÓN DE DOCENTE
VERSIÓN: 01
FECHA: 29-05-09



SISTEMA DE REGISTRO Y CONTROL DE NOTAS
DIAGRAMA DE CASOS DE USO
NIVEL: 2 GESTIÓN DE NOTAS
VERSIÓN: 01
FECHA: 29/05/09



3.1.3 DESCRIPCIÓN DE CASO DE USO

Nombre:	Agregar Alumno
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Guarda la información del alumno	
Actores: administrador	
Precondiciones: 1. Que no exista el alumno a ingresar	
Flujo normal: 1. Crea un carnet al alumno máximo de caracteres 5 2. Ingresa nombre del alumno máximo de caracteres 50 3. Ingresa apellidos del alumno máximo de caracteres 50 4. Selecciona el genero 5. Selecciona a que grado pertenece 6. Selecciona a que sección pertenece 7. Ingresa el teléfono 8. Ingresa el nombre del encargado 9. Ingresa la dirección de donde vive el alumno 10. Dar clic en el botón guardar	
Post condiciones: 1. Que guardo exitosamente la información del alumno	

Nombre:	Modificar Alumno
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Modifica la información del alumno	
Actores: administrador	
Precondiciones:	
1. Que exista la información del alumno a modificar	
Flujo normal:	
1. Ingresa el carnet del alumno 2. Dar clic en el botón buscar y aparecerá la información del alumno 3. Dar clic en el botón modificar y habilitara para la modificación de la información 4. Dar clic en el botón guardar	
Post condiciones:	
1. Que guardo exitosamente las modificaciones de la información	

Nombre:	Eliminar Alumno
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Elimina la información del alumno	
Actores: administrador	
Precondiciones:	
1. Que exista el carnet del alumno a eliminar	
Flujo normal:	
1. Ingresar el carnet del alumno a eliminar 2. Dar clic en buscar y aparecerá la información del alumno 3. Dar clic en eliminar y borrara la información del alumno	
Post condiciones:	
1. Que ha sido eliminado exitosamente la información	

Nombre:	Agregar Docente
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Guarda la información del docente	
Actores: administrador	
Precondiciones:	
1. Que no exista el docente a ingresar	
Flujo normal:	
1. Crea id al docente máximo de caracteres 6	
2. Ingresa nombre del docente máximo de caracteres 50	
3. Ingresa apellidos del docente máximo de caracteres 50	
4. Ingresa el teléfono	
6. Ingresa la dirección de donde vive el docente	
7. Dar clic en el botón guardar	
Post condiciones:	
1. Que guardo exitosamente la información	

Nombre:	Modificar Docente
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Modifica la información del docente	
Actores: administrador	
Precondiciones:	
1. Que exista la información del docente a modificar	
Flujo normal:	
1. Ingresa el id del docente 2. Dar clic en el botón buscar y aparecerá la información del docente 3. Dar clic en el botón modificar y habilitara para la modificación de la información. 4. Dar clic en el botón guardar	
Post condiciones:	
1. Que guardo exitosamente las modificaciones de la información del docente	

Nombre:	Eliminar Docente
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Elimina la información del docente	
Actores: administrador	
Precondiciones:	
1. Que exista el id del docente a eliminar	
Flujo normal:	
1. Ingresar el id del docente a eliminar 2. Dar clic en buscar y aparecerá la información del docente 3. Dar clic en eliminar y borrarla la información del docente	
Post condiciones:	
1. Que ha sido eliminado exitosamente la información	

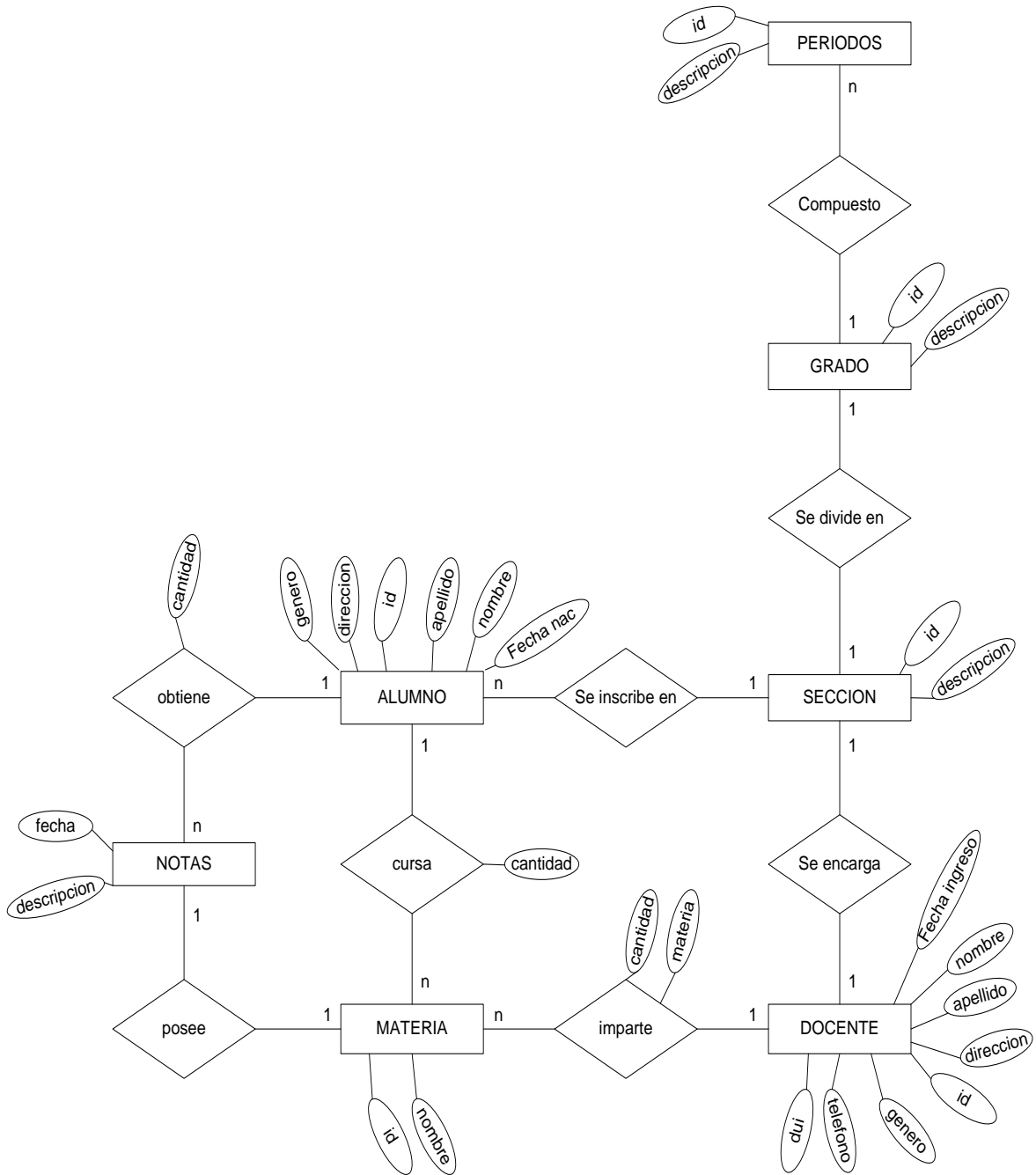
Nombre:	Agregar Nota
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Guarda las notas de los alumnos	
Actores: administrador, docente	
Precondiciones: 1. Que exista un carnet, grado, sección, materia y periodo	
Flujo normal: 1. Seleccionar grado, sección, materia, periodo y actividades a realizar 2. Ingresar actividades, autoevaluación, heteroevaluación y prueba objetiva 3. Dar clic en el botón guardar	
Post condiciones: 1. Que guardo exitosamente la información	

Nombre:	Modificar Nota
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Modifica las notas de los alumnos	
Actores: administrador, docente	
Precondiciones: 1. Que exista grado, sección y carnet	
Flujo normal: 1. Selecciona grado y sección 2. Modifica actividades, autoevaluación, heteroevaluación y prueba objetiva 3. Dar clic en el botón modificar y habilitar para la modificación de la información 4. Dar clic en el botón guardar	
Post condiciones: 1. Que guardo exitosamente las modificaciones de la información	

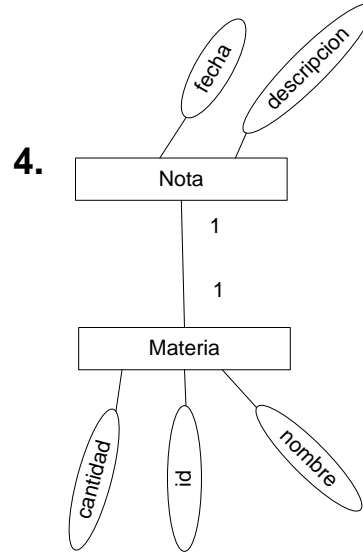
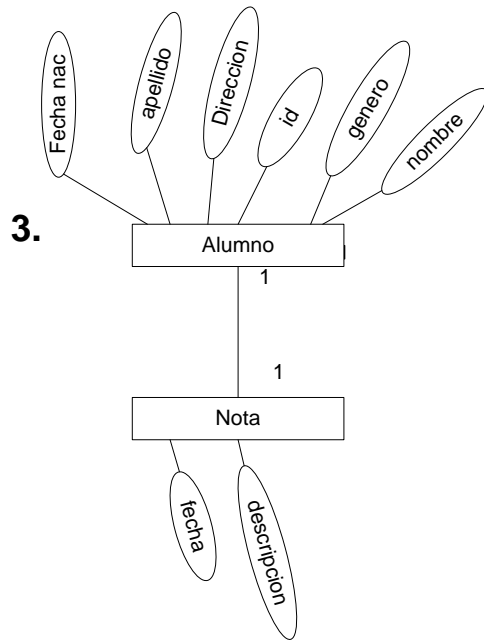
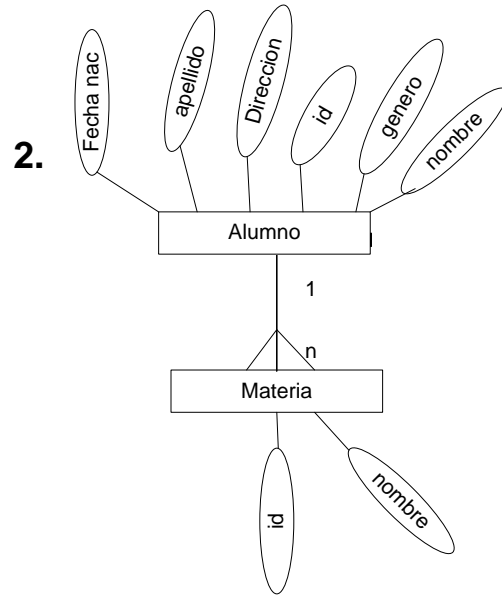
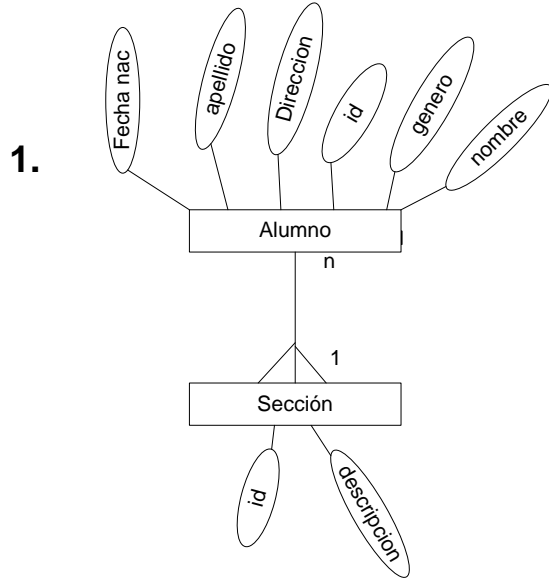
Nombre:	Eliminar Notas
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: Elimina la información del docente	
Actores: administrador, docente	
Precondiciones:	
1. Que exista carnet y notas	
Flujo normal:	
1. Ingresar el carnet del alumno 2. Dar clic en buscar y aparecerá las notas del alumno 3. Dar clic en eliminar y borrar las notas del alumno	
Post condiciones:	
1. Que ha sido eliminado exitosamente	

Nombre:	Consulta Notas
Autor:	William Alfredo Beltrán Najarro
Fecha:	01/10/09
Descripción: muestra la información del docente	
Actores: administrador, docente, alumno	
Precondiciones:	
1. Que exista carnet y notas	
Flujo normal:	
1. Ingresar el carnet del alumno 2. Dar clic en buscar y aparecerá las notas del alumno	
Post condiciones:	
1. Mostrar la materia y notas que el alumno a cursado	

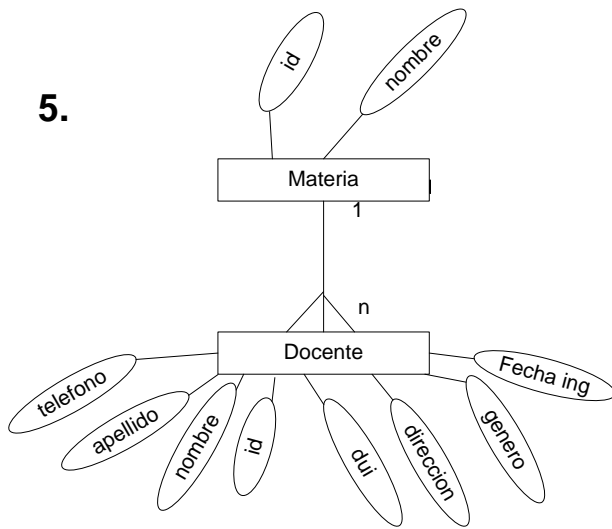
3.1.4 DIAGRAMA ENTIDAD RELACIÓN



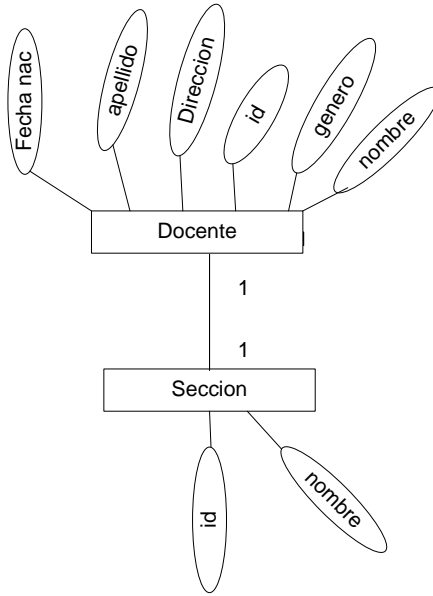
3.1.5 MAPEO



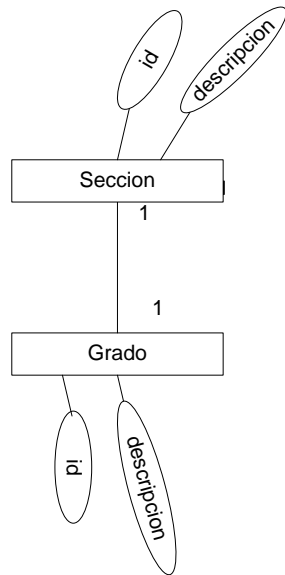
5.



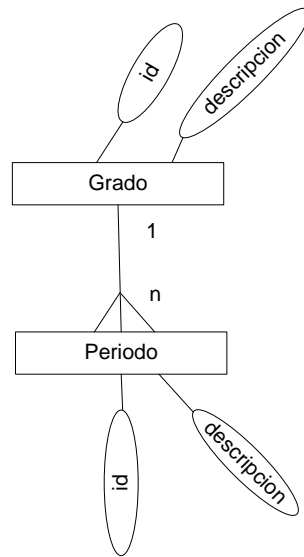
6.



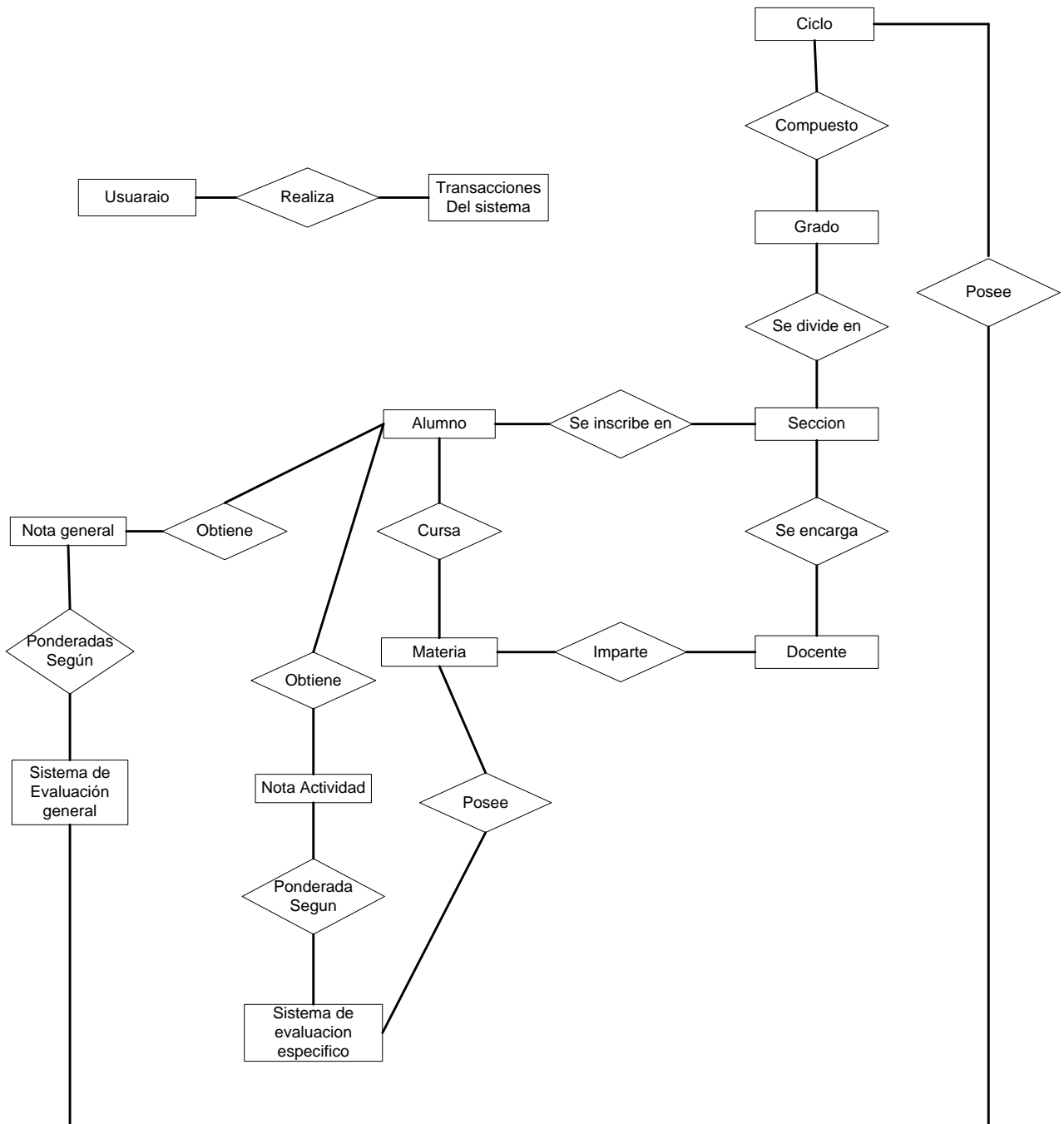
7.



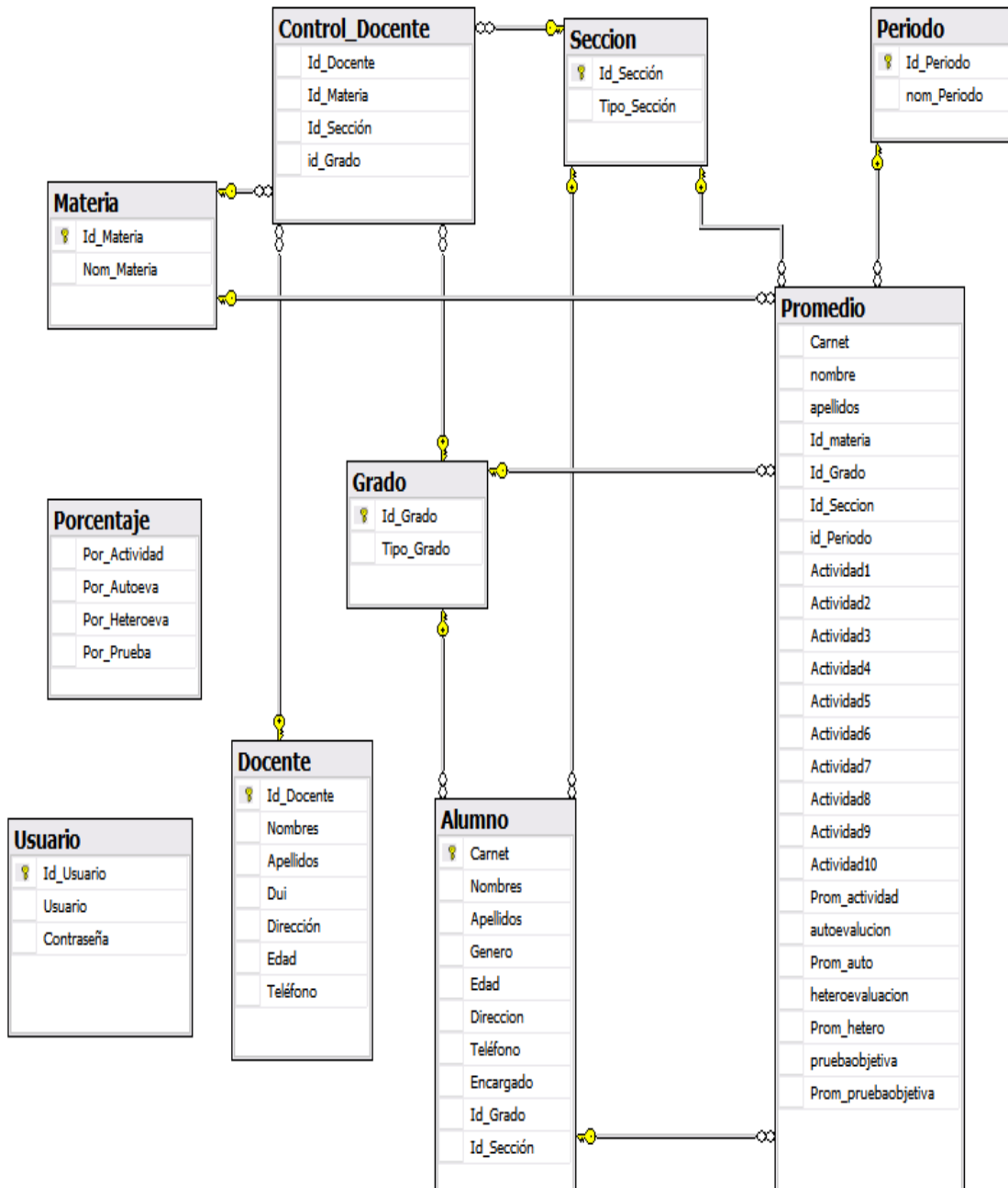
8.



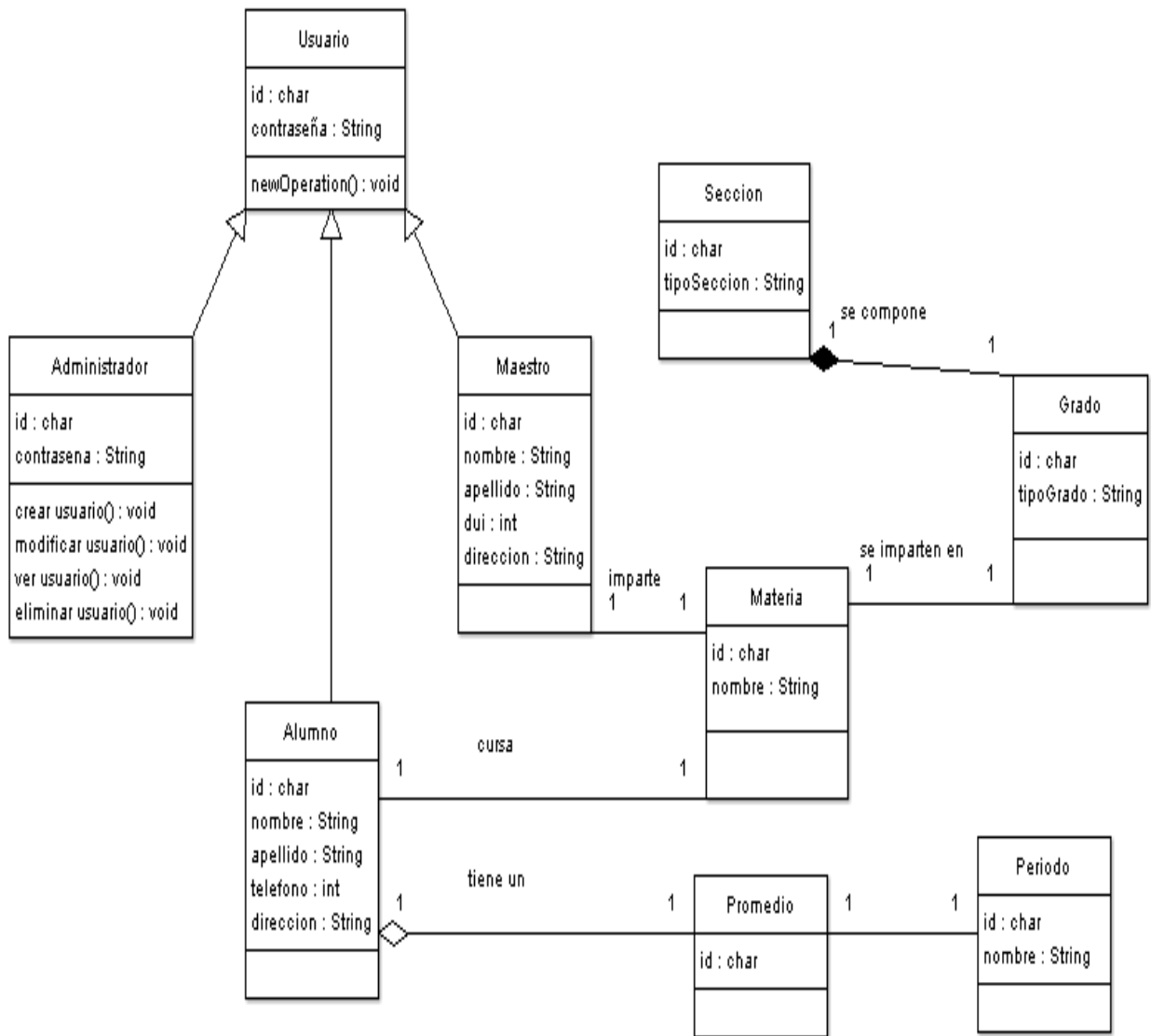
3.1.6 MODELO CONCEPTUAL



3.1.7 MODELO FÍSICO



3.1.8 DIAGRAMA DE CLASES DE EL SISTEMA DE CONTROL DE NOTAS



3.1.9 DICCIONARIO DE DATOS DEL SISTEMA.

Entidad **ALUMNO**: Esta tabla guardara los datos de cada alumno como son: carnet, nombres, apellidos, genero, edad, dirección, teléfono, encargado, grado, sección.

ALUMNO

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del alumno	Carnet	carácter/números	char	5	carnet del alumno
nombre del alumno	Nombres	A-Z	varchar	50	nombre del alumno
apellido del alumno	Apellidos	A-Z	varchar	50	apellido del alumno
género del alumno	Genero	masculino, femenino	nvarchar	15	sexo del alumno
edad del alumno	Edad	números	nvarchar	2	edad del alumno
dirección del alumno	Dirección	carácter/números	nvarchar	75	dirección del alumno
téléfono del alumno	Téléfono	números	nvarchar	9	téléfono del alumno
encargado del alumno	Encargado	A-Z	nvarchar	50	responsable del alumno
grado del alumno	Id_ grado	carácter/números	varchar	50	grado del alumno
sección del alumno	Id _Sección	carácter/números	varchar	4	sección del alumno

Entidad **DOCENTE**: contiene datos sobre el maestro.

DOCENTE

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del docente	Id_Docente	carácter/números	char	6	código del docente
nombre del docente	Nombres	A-Z	varchar	50	nombre del docente
apellido del docente	Apellidos	A-Z	varchar	50	apellido del docente
DUI del docente	Dui	carácter/números	nvarchar	15	DUI del docente
edad del alumno	Edad	números	nvarchar	2	edad del docente
dirección del alumno	Dirección	carácter/números	nvarchar	75	dirección del alumno
téléfono del alumno	Téléfono	números	nvarchar	9	téléfono del alumno

Entidad **CONTROL DOCENTE**: contiene información sobre el control de las materias, sección y el grado que imparte el docente.

CONTROL DOCENTE

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del docente	Id_Docente	carácter/números	varchar	6	código del docente
código de la materia	Id_Materia	carácter/números	varchar	8	código de la materia
código de la sección	Id_Sección	carácter/números	varchar	4	código de la sección
código del grado	Id_Grado	carácter/números	varchar	4	código del grado

Entidad **GRADO**: contiene datos sobre los grados.

GRADO

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del grado nombre del grado	Id_Grado Tipo_Grado	carácter/números A-Z	varchar varchar	4 40	código del grado nombre del grado

Entidad **MATERIA**: contiene información sobre las materias.

MATERIA

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código de la materia nombre de la materia	Id_Materia Nom_Materia	carácter/números carácter/números	varchar varchar	8 40	código de la materia nombre de la materia

Entidad **PERIODO**: contiene información sobre los periodos.

PERIODO

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del periodo nombre del periodo	Id_Periodo Nom_Periodo	carácter/números carácter/números	nchar varchar	2 10	código del periodo nombre del periodo

Entidad **PORCENTAJE**: contiene parámetros sobre las evaluaciones.

PORCENTAJE

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
porcentaje de actividad	por_Actividad	números	decimal	2,2	porcentaje de actividad
porcentaje de autoevaluación	por_Autoeva	números	decimal	2,2	porcentaje de autoevaluación
porcentaje de heteroevaluación	por_Heteroeva	números	decimal	2,2	porcentaje de heteroevaluación
porcentaje de prueba objetiva	por_Prueba	números	decimal	2,2	porcentaje de prueba objetiva

Entidad **SECCIÓN**: contiene información sobre las secciones.

SECCIÓN

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código de la sección tipo de sección	Id_Sección Tipo_Seccion	carácter/números carácter/números	varchar nchar	4 10	código de la sección tipo de sección

Entidad **PROMEDIO**: contiene información sobre las notas del alumno.

PROMEDIO

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del alumno	Carnet	carácter/números	char	5	carnet del alumno
nombre del alumno	Nombres	A-Z	varchar	50	nombre del alumno
apellido del alumno	Apellidos	A-Z	varchar	50	apellido del alumno
código de la materia	Id_materia	carácter/números	varchar	8	materias del alumno
código del grado	Id_grado	carácter/números	varchar	4	grado del alumno
código de la sección	Id_Sección	carácter/números	varchar	4	sección del alumno
actividad 1	Actividad1	números	varchar	4	nota 1 del alumno
actividad 2	Actividad2	números	varchar	4	nota 2 del alumno
actividad 3	Actividad3	números	varchar	4	nota 3 del alumno
actividad 4	Actividad4	números	varchar	4	nota 4 del alumno
actividad 5	Actividad5	números	varchar	4	nota 5 del alumno
actividad 6	Actividad6	números	varchar	4	nota 6 del alumno
actividad 7	Actividad7	números	varchar	4	nota 7 del alumno
actividad 8	Actividad8	números	varchar	4	nota 8 del alumno
actividad 9	Actividad9	números	varchar	4	nota 9 del alumno
actividad 10	Actividad10	números	varchar	4	nota 10 del alumno
promedio de actividad	Prom_Activ	números	varchar	4	promedio de actividad
autoevaluación	Autoevaluación	números	varchar	4	autoevaluación
Promedio autoevaluación	Prom_Auto	números	varchar	4	Promedio de autoevaluación
heteroevaluación	Heteroevaluación	números	varchar	4	heteroevaluación
Promedio heteroevaluación	Prom_hetero	números	varchar	4	Promedio heteroevaluación
prueba objetiva	Prueba_objetiva	números	varchar	4	prueba objetiva
Promedio prueba objetiva	Prom_Objetiva	números	varchar	4	promedio prueba objetiva
promedio final	Prom_final	números	varchar	4	promedio final

Entidad **USUARIO**: contiene información sobre los usuarios del sistema.

USUARIO

Nombre de campo	Nombre técnico	Valores permitidos	Tipo de campo	Longitud aprox.	Descripción
código del usuario	Usuario	carácter/números	nvarchar	10	código del usuario
contraseña del usuario	Contraseña	carácter/números	varchar	15	contraseña del usuario

3.2 NUEVAS FORMULACIONES TEÓRICAS

3.2.1 Beneficios y ventajas de la propuesta

- Facilitar la consulta de notas de los alumnos.
- Mejorar los procesos del control de notas por parte de los docentes
- Reducción de tiempo en las funciones de personal administrativo.
- Economizar recursos.

CONCLUSIONES

Resumiendo los resultados obtenidos en la investigación se puede concluir:

- sistema de registro de Control de notas del Centro Escolar agilizará y mejorará el proceso de actualización de la información.
- Se minimiza la posibilidad de extravió de información, al utilizar base de datos que garantizan que la información puede ser consultada en cualquier momento.
- Se logra mantener la integridad de la información relacionada a todo el alumnado del Centro Escolar.
- Se evita la posibilidad de recurrir a terceros en caso consulta de información de los alumnos por parte de las personas responsables de los mismos.

BIBLIOGRAFÍA

- Fundamentos de Programación (Tercera Edición) Autor: Luis Joyanes Aguilar.
- Base de datos con Visual Basic. Autor: Luis Durán.

Links:

- <http://es.wikipedia.org/wiki/.php>
- http://es.wikipedia.org/wiki/C_Sharp
- http://es.wikipedia.org/wiki/C_Sharp
- <http://urriellu.net/es/articles-software/csharp-advantages.html>
- http://www.americati.com/doc/ventajas_c/ventajas_c.html
- <http://www.monografias.com/trabajos/java/java.shtml>
- http://es.wikipedia.org/wiki/Visual_Basic
- <http://www3.uji.es/~mmarques/f47/apun/node83.html>
- http://www.neuronsrl.com.ar/training/uml/uml_intro.html
- http://es.wikipedia.org/wiki/Diagrama_de_Flujo_de_Datos
- <http://es.wikipedia.org/wiki/SQL>

ANEXOS