



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

INSTITUTO TECNOLÓGICO SUPERIOR CORDILLERA



ESCUELA DE SISTEMAS

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
TECNÓLOGO ANÁLISTA DE SISTEMAS.**

TEMA:

DESARROLLO E IMPLEMENTACION DE UN SISTEMA DE CONTROL
ACADÉMICO PARA EL COLEGIO “DR. MANUEL BENJAMÍN CARRIÓN MORA”

AUTOR:

DANIEL EFRAÍN GRANIZO CALVACHE

TUTOR:

TCNLO. OCTAVIO CÓNDR

Quito – Ecuador

ABRIL 2012



DECLARACION DE AUTENTICIDAD

El abajo firmante, declara que los contenidos y los resultados obtenidos en el presente proyecto, como requerimiento previo para la obtención del Título de Tecnólogo Analista de Sistemas, son absolutamente originales, auténticos, personales y de exclusiva responsabilidad legal y académica del autor.

DANIEL EFRAIN GRANIZO CALVACHE

1717833139



APROBACIÓN DEL TUTOR

En mi calidad de Tutor del trabajo sobre el tema: “SISTEMA DE CONTROL ACADÉMICO PARA EL COLEGIO DR. MANUEL BENJAMÍN CARRIÓN MORA”, presentado por el ciudadano: DANIEL EFRAIN GRANIZO CALVACHE, estudiante de la Escuela de Sistemas, considero que dicho informe reúne los requisitos y méritos suficientes para ser sometido a la evaluación por parte del Tribunal de Grado, que el Honorable Consejo de Escuela designe, para su correspondiente estudio y calificación.

Ing. Octavio Córdor

TUTOR



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

APROBACIÓN DEL TRIBUNAL DE GRADO

Los miembros del Tribunal de Grado designado por el Honorable Concejo de la Escuela de Sistemas, aprueban el trabajo de investigación de acuerdo con las disposiciones reglamentarias emitidas por el Centro de Investigaciones Tecnológicas y Proyectos del Instituto Tecnológico Superior Cordillera” para proyectos de grado de Tecnólogos Analistas de Sistemas: del Sr: DANIEL EFRAÍN GRANIZO CALVACHE

Para constancia firman:

PRESIDENTE

VOCAL1

VOCAL2



DEDICATORIA

En cada una de las letras de este proyecto va
Dedicado a Dios y a mi señora madre por estar siempre
apoyándome en las buenas y en las malas,
porque a pesar de mis errores, no dejo que me
rindiera, y siempre estuvo a mi lado.

DANIEL EFRAIN GRANIZO CALVACHE



AGRADECIMIENTO

Con gratitud a mis distinguidos maestros, que con nobleza y entusiasmo vertieron todo su apostolado en mi alma.

A mi querido Instituto Tecnológico Superior Cordillera, porque en sus aulas recibí los más gratos recuerdos que nunca olvidare.

A Dios que está en el cielo, por su valiosa Dirección.

A mis queridos compañeros y entrañables amigos, con quienes supe vencer las barreras que me puso el tiempo.

A mis padres, que con sus consejos valiosos supieron darme apoyo en cada momento.

DANIEL EFRAIN GRANIZO CALVACHE



PENSAMIENTO

Solo venciéndote,
Vencerás

DANIEL GRANIZO



RESUMEN EJECUTIVO

Capítulo I

En este capítulo podemos analizar de manera minuciosa el problema u objeto de estudio, aquí definimos los objetivos y el alcance que tendrá el proyecto.

Capítulo II

Conoceremos los antecedentes, reseña histórica de la institución en la cual se desarrollará el proyecto, también definiremos las herramientas que nos servirán para el desarrollo del mismo basándonos en las diferentes leyes que se aplicaran.

Capítulo III

Se aplica varios tipos y métodos de investigación para poder entender de una manera más clara la problemática, aplicando las distintas técnicas de investigación.

Capítulo IV

Esta parte es muy importante porque nos permite analizar y desarrollar en si todo el funcionamiento del sistema a desarrollarse,

Capítulo V

Es este capítulo vemos los diferentes impactos que tiene el proyecto, y así poder llegar a unas conclusión y recomendaciones más concretas.

Capítulo VI

Esta parte nos permite elaborar un presupuesto y cronograma de actividades a seguir durante el desarrollo del mencionado proyecto.



INDICE DE CONTENIDOS

CAPITULO I.....	1
EL PROBLEMA	1
1.1. Planteamiento Del Problema	1
1.2. Formulación Del Problema	2
1.3. Delimitación Del Problema	2
1.4. Objetivos.....	3
1.4.1 Objetivo General.....	3
1.4.2 Objetivos Específicos.....	3
1.5. Justificación E Importancia	3
1.5 Alcance	4
Módulo De Seguridad.....	4
Módulo De Mantenimientos.....	4
Módulo Transaccional	4
Módulo Administrativo O Secretaria.....	5
Reportes	5
Consultas	5
CAPITULO II.....	6
MARCO TEÓRICO	6
2.1. Antecedentes	6
2.2. Reseña Histórica.....	6
Visión.....	8
Misión.....	8
2.3. Marco Referencial	8
Microsoft Visual Studio	9
Lenguaje de Programación	10
Base De Datos	10
Toad Data Modeler 2.25l	11
Microsoft Project 2007	12
Microsoft Word 2010	12
Rational Rose	13



Microsoft Expression Blend 4	14
Microsoft Visio	14
Lenguaje Unificado de Modelado o UML	15
Metodología de desarrollo	16
Servidor De Base De Datos	16
2.4. Marco Legal.....	16
LEY DE LA PROPIEDAD INTELECTUAL.....	17
SECCIÓN II.....	17
OBJETO DEL DERECHO DE AUTOR	17
SECCIÓN V.....	18
DISPOSICIONES ESPECIALES SOBRE CIERTAS OBRAS.....	18
PARÁGRAFO PRIMERO	18
DE LOS PROGRAMAS DE ORDENADOR.....	18
LEY DE EDUCACIÓN	20
LEY Nº. 127 DEL 15 DE ABRIL DE 1983	20
LA CÁMARA NACIONAL DE REPRESENTANTES.....	20
EXPIDE LA SIGUIENTE LEY DE EDUCACIÓN.....	20
PRINCIPIOS GENERALES.....	20
Capítulo I	20
OBJETO DE LA LEY	21
Capítulo II	21
PRINCIPIOS Y FINES	21
Título Segundo	22
ESTRUCTURA DEL SISTEMA EDUCATIVO.....	22
Capítulo I	22
ESTRUCTURA GENERAL.....	22
Título Tercero.....	24
ORGANIZACIÓN Y ADMINISTRACIÓN DEL SISTEMA EDUCATIVO	24
Capítulo I	24
EL MINISTERIO DE EDUCACIÓN	24
Capítulo III	24
ESTRUCTURA DEL MINISTERIO.....	24



Capítulo V	24
DE LA ORIENTACIÓN Y BIENESTAR ESTUDIANTILES	24
Capítulo VII	25
INFRAESTRUCTURA EDUCATIVA	25
Título Cuarto	25
EL RÉGIMEN ESCOLAR	25
Capítulo I	25
EL AÑO LECTIVO, EL INGRESO Y LA EVALUACIÓN	25
Título Quinto.....	25
PERSONAL DOCENTE Y ADMINISTRATIVO	26
Capítulo I	26
EL MAGISTERIO NACIONAL	26
CAPÍTULO III.....	27
METODOLOGÍA DE INVESTIGACIÓN.....	27
3.1. Tipos De Investigación	27
3.1.1. Investigación Histórica	27
3.1.2. Investigación Bibliográfica Documental.....	27
3.1.3. Investigación Descriptiva	28
3.1.4. Investigación Explicativa	29
3.2. Métodos Empleados en la Investigación	29
3.2.1. Método Inductivo	29
3.2.2. Método Deductivo.....	29
3.2.3. Método Histórico Lógico	30
3.2.4. Método Analítico Sintético	30
3.2.5. Método Sistémico.....	31
3.3. Técnicas De Recolección De Información.....	31
3.3.1. Observación	31
3.3.2. Entrevista	31
3.3.3. Encuesta.....	33
CAPITULO IV	40
DESARROLLO DE LA PROPUESTA	40
4.1. Diagnostico Situacional.....	40



4.2. Estructura Organizacional	41
¿Qué es organizar?	42
4.2.1. Orgánico Estructural	42
4.2.2. Orgánico Funcional	42
4.3. Infraestructura Informática	44
4.3.1. Hardware	44
4.3.2. Software.....	45
4.3.3. Comunicaciones.....	46
4.3.3.1. Topología en estrella:	47
Ventajas	48
Desventajas	48
4.3.4. Recurso Humano Técnico.....	48
4.4. Descripción De Alternativas	48
4.4.1. Alternativa 1: Contratar A La Empresa Sidesoft Cia. Ltda Para El Desarrollo Del Software	49
4.4.2. Alternativa 2: Contratar A La Empresa Logic Studio S.A Para El Desarrollo Del Software.	50
4.4.3. Alternativa 3: Proyecto De Tesis.....	51
4.5. Evaluación Y Selección De Alternativas	52
4.5.1. Evaluación Técnica.....	53
4.5.2. Evaluación De Garantía Económica	54
4.5.3. Evaluación De Garantía Técnica	54
4.5.4. Evaluación De Soporte Técnico	54
4.6. Factibilidad Técnica	55
4.7. Descripción De Procesos	56
LOGIN	56
INSCRIPCIONES	56
MATRICULA.....	56
NOTAS	57
4.8. Descripción De Metodología De Desarrollo	57
4.8.1. Metodología de desarrollo aplicada	58
4.8.1.1. Introducción al RUP	58



4.8.1.2. Dimensiones del RUP.....	59
4.8.1.3. Fases del RUP	61
4.8.1.4. Disciplinas y Flujos de Trabajo del RUP.....	64
4.8.1.4.1. Modelado del negocio.....	65
4.8.1.4.2. Requerimientos.....	65
4.8.1.4.3. Análisis y diseño.....	65
4.8.1.4.4. Implementación	65
4.8.1.4.5. Pruebas.....	66
4.8.1.4.6. Despliegue	66
4.8.1.4.7. Gestión y configuración de cambios.....	66
4.8.1.4.8. Gestión del proyecto	67
4.8.1.4.9. Entorno.....	67
4.9. Documento De Especificación De Requerimientos	67
4.10. Modelo Físico.....	68
4.11. Diccionario De Datos.....	68
4.12. Estándares De Programación.....	68
4.12.1. Estándares de Base De Datos	69
4.12.2. Estándares de Diseño.....	71
4.12.3. Estándares de Programación.....	72
4.13. Documento De Diseño	73
4.14. Pruebas Y Depuración	73
4.14.1. Pruebas de Integración	73
4.14.2. Pruebas de Validación	74
4.14.3. Pruebas de Seguridad	75
4.14.4. Pruebas de interfaces graficas de usuario.....	76
4.15. Instalación Del Sistema	76
4.16. Recopilación Y Carga De Datos	76
4.17. Pruebas Y Depuración Final En Funcionamiento.....	77
4.18. Puesta En Marcha Del Sistema	77
4.19. Capacitación Al Usuario Final	77
4.20. Capacitación Al Personal Técnico.....	79
CAPITULO V	81



ANÁLISIS DE IMPACTOS.....	81
5.1. Principales Impactos.....	81
5.1.1. Científico.....	81
5.1.2. Educativo.....	81
5.1.3. Técnico.....	81
5.1.4. Tecnológico.....	82
5.1.5. Empresarial.....	82
5.1.6. Social.....	82
5.1.7. Económico.....	83
5.2. Conclusiones.....	83
5.3. Recomendaciones.....	84
CAPITULO VI.....	85
6.1. Cronograma de Actividades.....	85
6.2. Glosario de Términos.....	85
6.3. Bibliografía.....	99
6.4. Netgrafía.....	99



ÍNDICE DE GRÁFICOS

Gráfico N° 1 Ubicación del Colegio	2
Gráfico N° 2 Interfaz de Visual Studio 2008	9
Gráfico N° 3 Interfaz de SQL Server 2008	11
Gráfico N° 4 Interfaz de Modelador de BDD Toad	11
Gráfico N° 5 Interfaz de Project 2007.....	12
Gráfico N° 6 Interfaz de Word 2010	13
Gráfico N° 7 Interfaz de Rational Rose	13
Gráfico N° 8 Interfaz de Microsoft Expression Blend 4	14
Gráfico N° 9 Interfaz de Visio 2007	15
Gráfico N° 10 Resultados Pregunta 1	35
Gráfico N° 11. Resultados Pregunta 2	36
Gráfico N° 12. Resultados Pregunta 3	37
Gráfico N° 13. Resultados Pregunta 4	38
Gráfico N° 14. Resultados Pregunta 5	38
Gráfico N° 15. Resultados Pregunta 6	39
Gráfico N° 16. Disciplinas, fases, iteraciones del RUP	60
Gráfico N° 17. Fases, iteraciones y flujo de trabajo del RUP	62
Gráfico N° 18. Disciplinas y flujo de trabajo del RUP	64



ÍNDICE DE TABLAS

Tabla N° 1. Software a Utilizar.....	9
Tabla N° 2. Población del Colegio	35
Tabla N° 3. Resultados Pregunta 1.....	35
Tabla N° 4. Resultados Pregunta 2.....	36
Tabla N° 5. Resultados Pregunta 3.....	37
Tabla N° 6. Resultados Pregunta 4.....	37
Tabla N° 7. Resultados Pregunta 5.....	38
Tabla N° 8. Resultados Pregunta 6.....	39
Tabla N° 9. Infraestructura de Hardware	45
Tabla N° 10. Infraestructura de Software.....	46
Tabla N° 11. Especificaciones Técnicas.....	49
Tabla N° 12. Propuesta alternativa 1	50
Tabla N° 13. Propuesta alternativa 1	51
Tabla N° 14. Propuesta Proyecto de Tesis.....	52
Tabla N° 15. Evaluación de especificaciones técnicas.	53
Tabla N° 16. Evaluación de garantía económica	54
Tabla N° 17. Evaluación de garantía técnica.....	54
Tabla N° 18. Evaluación soporte técnico	55
Tabla N° 19. Estándar de base de datos para tablas.	69
Tabla N° 20. Estándar de tablas utilizadas	70
Tabla N° 21. Estándar de un atributo de una tabla.....	71
Tabla N° 22. Estándar de diseño.	71
Tabla N° 23. Ejemplo de variables.	72
Tabla N° 24. Estándar de tipo de datos utilizados	72
Tabla N° 25. Prueba de integración.	74
Tabla N° 26. Prueba de validación.	74
Tabla N° 27. Pruebas de Seguridad	75
Tabla N° 28. Ejemplo de Pruebas de Seguridad	75
Tabla N° 29. Pruebas de interfaz gráfica.	76
Tabla N° 30 Actividades Capacitación a Personal Técnico.....	80



ÍNDICE DE ANEXOS

Anexo N° 1. Formato de encuesta	102
Anexo N° 2. Diagrama Organizacional del Colegio Dr. M.B.C.M	104
Anexo N° 3. Diagrama Orgánico Funcional del Colegio Dr. M.B.C.M.....	106
Anexo N° 4. Diagrama de red del Colegio Dr. M.B.C.M.....	108
Anexo N° 5. Diagrama de estado de flujo de login.....	110
Anexo N° 6. Diagrama de estado de flujo de inscripción.....	112
Anexo N° 7. Diagrama de estado de flujo de matriculación.....	114
Anexo N° 8. Diagrama de estado de flujo de notas.....	116
Anexo N° 9. Documento SRS.....	118
Anexo N° 10. Modelo Físico.....	134
Anexo N° 11. Diccionario de datos.....	136
Anexo N° 12. Documento SDS.....	143
Anexo N° 13. Manual técnico.....	174
Anexo N° 14. Cronograma de actividades.....	188



CAPITULO I

EL PROBLEMA

1.1. Planteamiento Del Problema

Para el desarrollo de este proyecto se realizó un estudio en el colegio “Dr. Manuel Benjamín Carrión Mora”, en los procesos que realiza como en lo académico y administrativo, el cual dio como resultado de investigación que la institución no cuenta con una centralización de archivos, lo que da a entender que no cuenta con un sistema de control académica.

Con el estudio realizado se logró obtener que la información que tiene el colegio como registros de inscripciones, matriculas, notas, datos del personal docente, son guardados en archivos Excel, la cual es impresa, estos datos físicos son almacenadas en carpetas físicas, siendo así vulnerable a pérdidas o al deterioro de la misma.

Al tener los datos almacenados de manera digita (Excel) y en dos ordenadores estos se modifican constantemente, teniendo inconvenientes ya que no hay información en tiempo real, esto quiere decir que los datos de una computadora con la de la otra no eran iguales, teniendo mayor inconvenientes en las calificaciones de los estudiantes las cuales eran las que se modificaban cada momento.



INSTITUTO TECNOLÓGICO SUPERIOR CORDILLERA

Al momento de las inscripciones y matriculas no hay una integridad de información ya que el código de estos son los que se duplicaban, esto se debe a que el colegio no tienen un sistema automatizado que genere un código de manera automática para cada alumno.

Cabe mencionar que en el proceso de gestión académica hay demora en la entrega de reportes de información, siendo el tiempo un factor imposible de recuperarlo.

1.2. Formulación Del Problema

¿Con la implementación de un sistema de control académico se podrá dar solución a los diferentes inconvenientes que tiene el colegio en el proceso de gestión académica?

1.3. Delimitación Del Problema

El presente proyecto se va llevar a cabo en la provincia de Pichincha ciudad de Quito sector Norte en el Barrio Comité del Pueblo No. 1 en el Colegio Dr. Manuel Benjamín Carrión Mora el cual está ubicado en las calles Adolfo Klinger y Enrique Garcés, este proyecto será aplicado en el departamento de secretaria.



Gráfico N° 1
Ubicación del Colegio.
Fuente: Internet.



1.4. Objetivos

1.4.1 Objetivo General

Diseñar, desarrollar e implementar un Sistema de Control Académico para el Colegio Dr. Manuel Benjamín Carrión Mora.

1.4.2 Objetivos Específicos

- Conocer las reglas de negocio de la institución.
- Modelar la base de datos del sistema a implantarse.
- Desarrollar el código fuente del aplicativo, que va ser implantado en el colegio.
- Implementar y capacitar a los usuarios que manejaran el Sistema Informático.

1.5. Justificación E Importancia

El sistema a implantarse no tendrá ningún costo para la institución ya que no invertirá ningún valor monetario para el desarrollo del mismo por tratarse de un proyecto de grado.

Con la implementación del Sistema de Control Académico el colegio tendrá almacenada de manera confiable y segura la información, así no tendrá inconveniente que la información se pierda, lo que puede ser perjudicial no solo para el colegio sino también para los estudiantes.

El desarrollo de este proyecto es de vital importancia para la institución educativa ya que al momento de automatizar los procesos el usuario minimiza tiempo y se le facilita la manipulación y organización de la información obteniendo así una integridad de información para poder brindar una buena atención a los usuarios,



1.5 Alcance

Después de analizar y recopilar datos en el Colegio “DR. MANUEL BENJAMÍN CARRIÓN MORA” y en base a las necesidades, requerimientos y se plantea la estructura del sistema de la siguiente manera.

Módulo De Seguridad

Este módulo tendrá información de los usuarios que ingresen al sistema ya que podrán hacerlo mediante nombre de usuario y una contraseña.

Módulo De Mantenimientos

En este módulo se podrá editar, guardar, modificar y eliminar información referente al proceso de gestión académica tales como:

- Periodo
Contendrá información de cada periodo que tenga el colegio desde la implementación del sistema
- Jornadas
Se registrará la información de cada jornada que tenga el colegio.
- Especialidades
Se registrará la información de todas las especialidades que haya.
- Curso o Niveles
Contendrá información de cada curso.
- Registro de materias
Se registrara las materias que se dicten por cada curso.
- Alumnos
Se tendrá información de todo el alumnado.
- Profesores
Se registrara información referente al docente.

Módulo Transaccional



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

Este módulo es muy importante ya que aquí es donde esta las reglas de negocio del colegio.

Módulo Administrativo O Secretaria

- Inscripciones
En esta instancia se almacenara información de los alumnos aspirantes a los diferentes cursos o niveles.
- Matriculas
Se almacenara información de los alumnos matriculados en los cursos o niveles.
- Ingreso de alumnos
Se registrara los datos personales del alumnado desde el primer curso hasta sexto curso.
- Ingreso de notas
Contendrá la información de notas de todo el alumnado del colegio.
- Asignación materias por docentes
Se registrara la información de la asignación de materias que tiene cada docente a su cago.
- Registro de asistencia
Se registrara la información de asistencia de cada docente y alumnos.

Reportes

En este módulo el usuario podrá generar reportes de todas las inscripciones, matrículas y notas de todos los estudiantes y de manera específica.

Consultas

En este módulo se podrá realizar varias consultas en cuanto a la información académica del alumnado.



CAPITULO II

MARCO TEÓRICO

2.1. Antecedentes

La evolución de la tecnología ha variado mucho a lo largo del tiempo. Desde los primeros hombres hasta nuestros días la tecnología ha evolucionado para facilitarnos algunas tareas.

Por este motivo es que el Colegio “DR. MANUEL BENJAMÍN CARRIÓN MORA” desde la fecha de su creación hasta hoy ha crecido en su innovación informática, lo cual ha llevado a que los procesos que antes se realizaban manualmente hoy se dificulten en cuanto a la organización de la información, ocasionado así más tiempo en los procesos que realiza la institución con respecto a la gestión académica.

Por tal razón el colegio ha visto la necesidad de implantar un sistema que permita la gestión académica. Para de esta manera automatizar y facilitar los procesos.

El Colegio Nacional Dr. Manuel Benjamín Carrión Mora es una institución educativa la cual está dedicada a hacer de los jóvenes unos estudiantes y bachilleres técnicos.

2.2. Reseña Histórica

“Trabajar para vivir y vivir para trabajar”, es lema de muchas comunidades, en especial de esta, la del Comité del Pueblo N° 1 que se ha caracterizado por su



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

lucha tesonera por el bien de los más pobres. Es así como nace la institución que hoy se llama “COLEGIO NACIONAL TECNICO MIXTO DR. MANUEL BENJAMIN CARRION MORA”.

Se creó en el año de 1977-1978 cuando el Dr. Carlos Rodríguez Paredes, fungía Presidente de la Cooperativa de Vivienda “COMITÉ DEL PUEBLO” y con la visión del Prof. Horacio Tinajero, Dr. Carlos Zapata y otros dirigentes, ya que en el sector indicado el asentamiento de población era rápido y abundante lo que implicaba la existencia de una gran población estudiantil de nivel primario y medio; lo cual se hace efectivo a partir de Septiembre de 1977 con la recepción de las inscripciones luego de las matrículas, iniciando sus labores a partir del 2 de octubre del mismo año con el Primer Curso de Ciclo Básico y con el nombre de “Unidad Educativa Comité del Pueblo”, Sección Vespertina.

En el año 1978-1979, el Ministerio de Educación da nombramiento a los Profesores de Planta para que legalice su existencia; consiguiendo el funcionamiento de Primero y Segundo Curso Ciclo Básico.

El 3 de julio de 1979 se designa al Colegio con el nombre del ilustre hombre de letras ecuatoriano “DR. MANUEL BENJAMIN CARRION MORA” por parte del Ministerio de Educación, inmortalizando así a quien desempeñara altas funciones, tanto en la cátedra como en los campos literario, político y periodístico, cooperando de manera acertada en la formación de la niñez y juventud ecuatorianas, siendo un ejemplo su espíritu de lucha incansable en las conquistas sociales para las nuevas generaciones.

El Colegio cuenta con Personal Docente Especializado, ubicado conforme sus Títulos Académicos, en las diferentes Áreas de Estudio.

La presencia del Colegio Nacional Técnico Mixto "DR. MANUEL BENJAMÍN CARRIÓN MORA", es la respuesta al clamor de una comunidad pujante, luchadora, solidaria y en constante superación del Comité del Pueblo; sus inicios fueron duros, pero la tenacidad, la vocación de Maestros y mística de servicio, ha



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

permitido que hoy se convierta en Institución Educativa Referencial de la Educación Técnica de Pichincha.

Se encuentra inmerso en el PROYECTO DE REFORZAMIENTO DE LA EDUCACION TECNICA, siendo uno de los seis establecimientos beneficiario del PROYECTO DE AULAS DE MICRO ENSEÑANZA a nivel nacional.

Su trayectoria permite que se considere como un Colegio Inteligente y se proyecte a futuro como un Plantel representativo de la Educación Técnica con calidad y excelencia a nivel no solo provincial, nacional, sino internacional.

Visión

Tenemos que ser un pueblo grande en los hábitos de la espiritualidad, de la ética, de la cultura y libertad.

Misión

Ofertar Octavos, Noveno y Décimo de Educación Básica, Bachillerato en Ciencias y Bachilleratos Técnicos Industriales, con Personal calificado, en un ambiente humano, con valores y mística de servicio.

2.3. Marco Referencial

Para el desarrollo de sistema de control académico se ha tomado en cuenta los siguientes software y metodologías de desarrollo a utilizar como se detalla en la siguiente tabla.

SOFTWARE A UTILIZAR		
HERRAMIENTAS	DESCRIPCIÓN	VERSION
Visual Studio 2010	IDE de Programación	2010
C#	Lenguaje de Programación	9.0
SQL EXPRESS 2008	Motor de BDD	10.0
Toad Data Modeler	Modelador BDD	2.25
Microsoft Project 2007	Herramienta administración	2007

	de proyectos	
Microsoft Word	Editor de texto	2010
Rational Rose	Diseñador de software	2002
Microsoft Expression Blend	Diseñador de interfaz grafica	4
Microsoft Visio 2007		2007

Tabla N° 1.

Software a Utilizar

Fuente: Internet

Microsoft Visual Studio

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

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

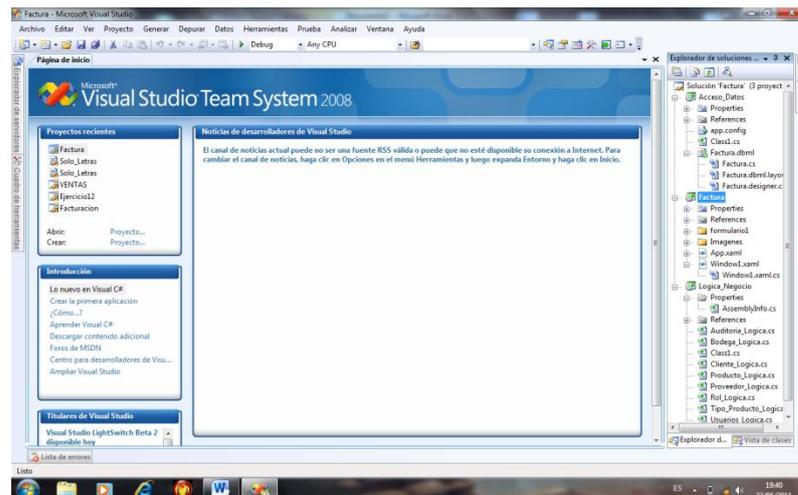


Gráfico N° 2

Interfaz de Visual Studio 2008

Fuente: Microsoft



Lenguaje de Programación

C# (leído en inglés “C Sharp” y en español “C Almohadilla”) es un lenguaje de programación que toma las mejores características de lenguajes preexistentes como Visual Basic, Java o C++ y las combina en uno solo. El hecho de ser relativamente reciente no implica que sea inmaduro, pues Microsoft ha escrito la mayor parte de la BCL usándolo, por lo que su compilador es el más depurado y optimizado de los incluidos en el *.NET Framework SDK*.

Aunque es posible escribir código para la plataforma .NET en muchos otros lenguajes, C# es el único que ha sido diseñado específicamente para ser utilizado en ella, por lo que programarla usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes ya que C# carece de elementos heredados innecesarios en .NET. Por esta razón, se suele decir que C# es el lenguaje nativo de .NET

Base De Datos

Microsoft SQL Server 2008 Express es un sistema de administración de datos eficaz y confiable que ofrece un variado conjunto de características, protección de datos y rendimiento para clientes de aplicaciones incrustadas, aplicaciones web ligeras y almacenes de datos locales. SQL Server 2008 Express, que está diseñado para una implementación sencilla y una creación de prototipos rápida, está disponible de forma gratuita y su redistribución con aplicaciones también es gratuita. Está diseñado para integrarse a la perfección con otras inversiones de infraestructura de servidor.

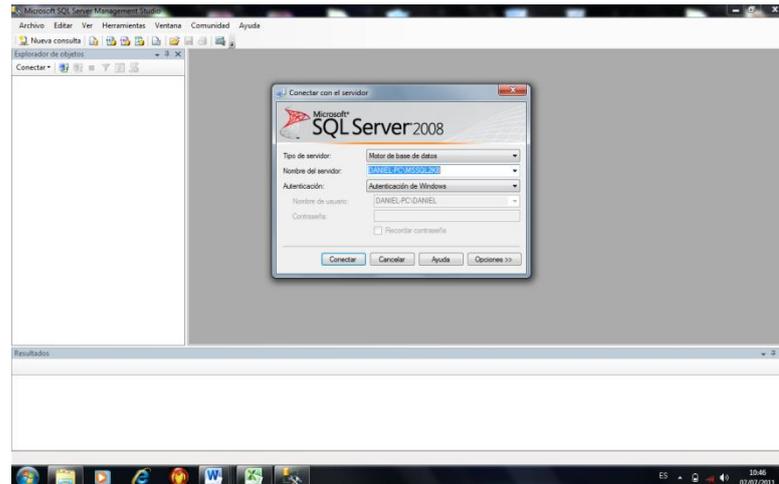


Gráfico N° 3
Interfaz de SQL Server 2008
Fuente: Microsoft

Toad Data Modeler 2.25I

Este software es un generador de tablas con sus diferentes campos para elaborar una base de datos que nos permite realizar un módulo de presentación sobre una base de datos este nos permite relacionar tablas y describir campos principales como secundarios el cual nos servirá de mucho para realizar nuestro primer prototipo del sistema del proyecto.

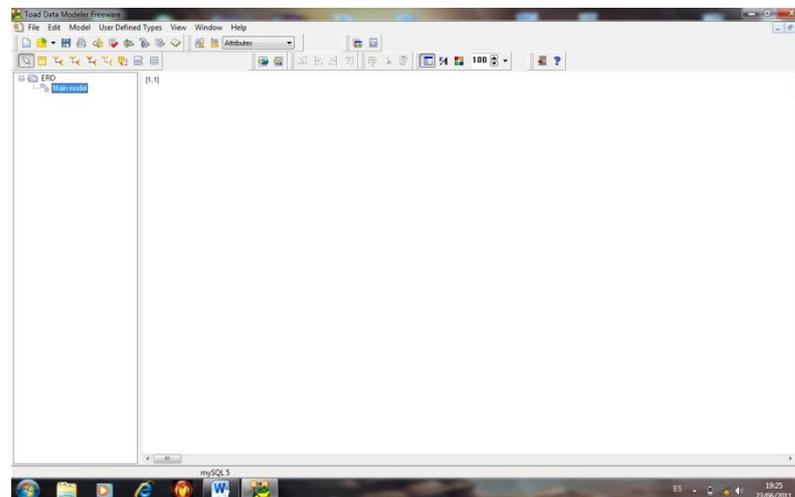


Gráfico N° 4
Interfaz de Modelador de BDD Toad
Fuente: Toad Data Modeler 2.25



Microsoft Project 2007

Microsoft Office Project Standard 2007 ofrece unas sólidas herramientas de administración de proyectos con la dosis adecuada de funcionalidad, potencial y flexibilidad, con el fin de administrar los proyectos con mayor eficacia y eficiencia. Podrá mantenerse informado y controlar el trabajo, la programación y las finanzas del proyecto, mantener la sintonía entre los equipos de proyecto y mejorar la productividad gracias a la integración con los conocidos programas del sistema Microsoft Office, las eficaces opciones de elaboración de informes, el planeamiento asistido y las herramientas flexibles.

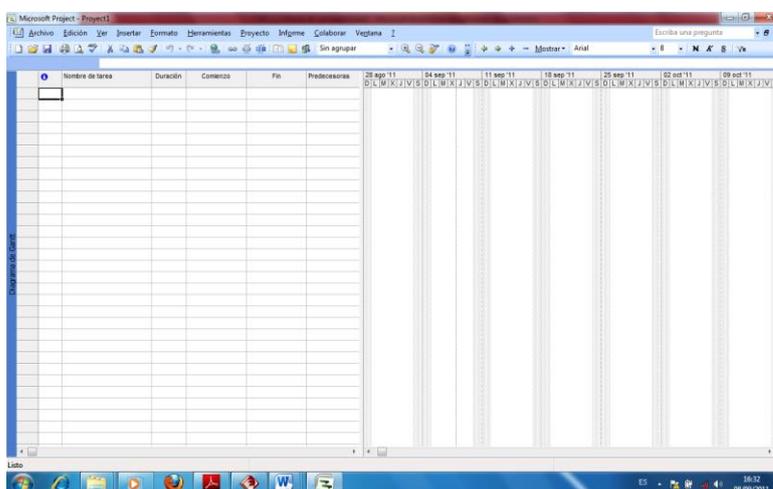


Gráfico N° 5
Interfaz de Project 2007
Fuente: Microsoft

Microsoft Word 2010

Microsoft Word, es un programa editor de texto ofimático muy popular, que permite crear documentos sencillos o profesionales. Microsoft Word es incluido en el paquete Microsoft Office de Microsoft.

Microsoft Word posee herramientas de ortografía, sinónimos, gráficos, modelado de texto, etc. Además tiene una muy buena integración con el resto de las aplicaciones del paquete Microsoft Office. Fue inicialmente desarrollado por Richard Brodie bajo el nombre de Multi-Tool Word para el ordenador de IBM, bajo

el sistema operativo DOS en el año 1983. Luego se crearon versiones para Apple Macintosh en 1984 y para Microsoft Windows en 1989.

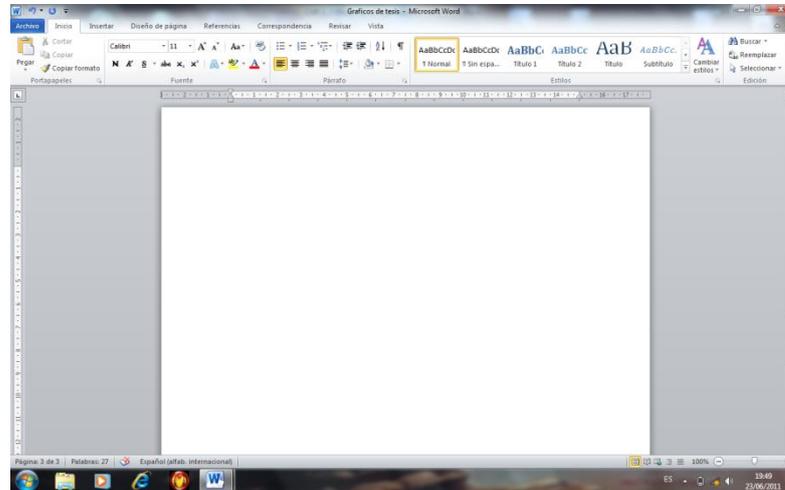


Gráfico N° 6
Interfaz de Word 2010
Fuente: Microsoft

Rational Rose

Es actualmente conocida como una familia de software de IBM para el despliegue, diseño, construcción, pruebas y administración de proyectos en el proceso desarrollo de software. Rational Machines fue fundada por Paul Levy y Mike Devlin en 1981 para proporcionar herramienta que expandieran las prácticas modernas de ingeniería de software, particularmente la arquitectura modular y el desarrollo iterativo.

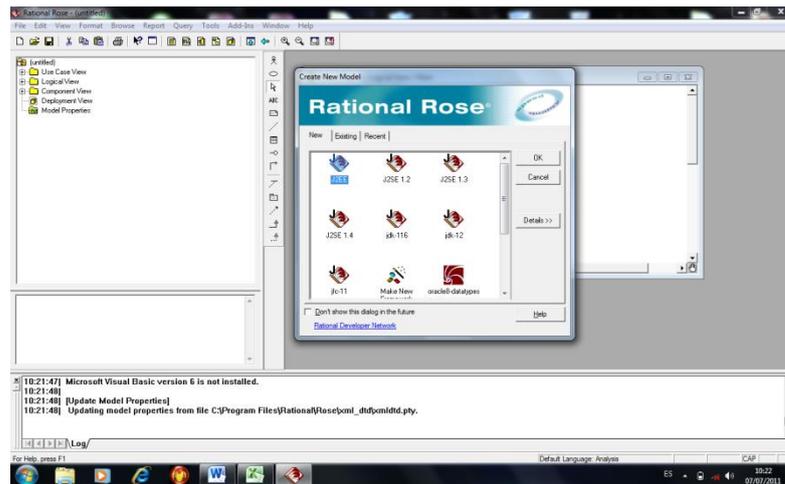


Gráfico N° 7
Interfaz de Rational Rose
Fuente: Microsoft

Microsoft Expression Blend 4

Expression Blend (antes llamado Expression Interactive Designer, cuyo nombre en código era Sparkle) es una herramienta profesional desarrollado por Microsoft, de diseño que le permite controlar la eficacia del XAML, .NET y Silverlight para proporcionar experiencias de usuario atractivas en escritorios conectados y Web para Windows, abriendo la puerta a la creación de diseños de interfaces de usuario mediante una amplia gama de tipos de medios, trabajando en un entorno de diseño en tiempo real.

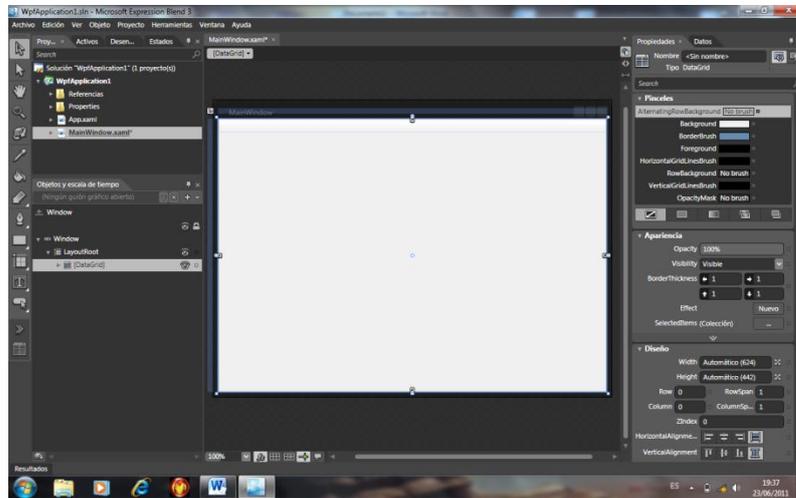


Gráfico N° 8
Interfaz de Microsoft Expression Blend 4
Fuente: Microsoft

Microsoft Visio

Es un software de dibujo vectorial para Microsoft Windows. Visio comenzó a formar parte de los productos de Microsoft cuando fue adquirida la compañía Visio en el año 2000.

Las herramientas que lo componen permiten realizar diagramas de oficinas, diagramas de bases de datos, diagramas de flujo de programas, UML, y más, que permiten iniciar al usuario en los lenguajes de programación.

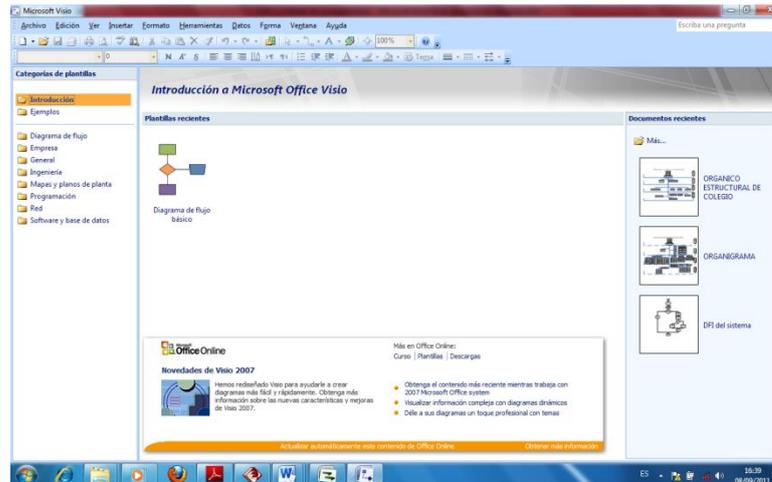


Gráfico N° 9
Interfaz de Visio 2007
Fuente: Microsoft

Lenguaje Unificado de Modelado o UML

Por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Es importante resaltar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.



Metodología de desarrollo

El Proceso Unificado de Rational (Rational Unified Process en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

Servidor De Base De Datos

Un servidor de base de datos es un programa que provee servicios de base de datos a otros programas u otras computadoras, como es definido por el modelo cliente-servidor. También puede hacer referencia a aquellas computadoras (servidores) dedicadas a ejecutar esos programas, prestando el servicio.

Los sistemas de administración de base de datos generalmente proveen funcionalidades para servidores de base de datos, en cambio otros (como por ejemplo, MySQL) solamente proveen construcción y acceso a la base de datos.

La mayoría de los servidores de bases de datos ofrecen una interfaz de texto rudimentaria que permite interactuar con el servidor usando SQL (Structured Query Language). La mayoría de los usuarios interactúan con la base de datos con software de cliente personalizado para utilizar la base de datos como un sistema para almacenar información

2.4. Marco Legal

La Comisión de Legislación y Codificación del H. Congreso Nacional de conformidad con la Constitución Política de la República, considera, realizar la Codificación de la Ley de Propiedad Intelectual, observando las disposiciones de la Constitución Política de la República; así como las reformas expresas, que se han producido en las leyes reformativas a esta Ley; Ley de Educación Superior; Código de Procedimiento Civil; Ley para la Transformación Económica del Ecuador; Código



de Procedimiento Penal; Ley Orgánica de Aduanas y Resolución No. 161-2000-TP, del Tribunal Constitucional.

El IEPI es la institución encargada en el Ecuador de registrar la propiedad intelectual en este caso un sistema de información es de propiedad de quien lo ha creado (De conformidad con el Art. 7 de la Ley de Propiedad Intelectual, autor es la persona natural que realiza la creación intelectual, no empresas, fundaciones, organizaciones u otros similares)

Art. 7.- Para los efectos de este Título los términos señalados a continuación tendrán los siguientes significados:

Programa de ordenador (software): Toda secuencia de instrucciones o indicaciones destinadas a ser utilizadas, directa o indirectamente, en un dispositivo de lectura automatizada, ordenador, o aparato electrónico o similar con capacidad de procesar información, para la realización de una función o tarea, u obtención de un resultado determinado, cualquiera que fuere su forma de expresión o fijación. El programa de ordenador comprende también la documentación preparatoria, planes y diseños, la documentación técnica, y los manuales de uso.

LEY DE LA PROPIEDAD INTELECTUAL

SECCIÓN II

OBJETO DEL DERECHO DE AUTOR

Art. 8. La protección del derecho de autor recae sobre todas las obras del ingenio, en el ámbito literario o artístico, cualquiera que sea su género, forma de expresión, mérito o finalidad. Los derechos reconocidos por el presente Título son independientes de la propiedad del objeto material en el cual está incorporada la obra y su goce o ejercicio no están supeditados al requisito del registro o al cumplimiento de cualquier otra formalidad.

Las obras protegidas comprenden, entre otras, las siguientes:



Libros, folletos, impresos, epistolarios, artículos, novelas, cuentos, poemas, crónicas, críticas, ensayos, misivas, guiones para teatro, cinematografía, televisión, conferencias, discursos, lecciones, sermones, alegatos en derecho, memorias y otras obras de similar naturaleza, expresadas en cualquier forma;

Colecciones de obras, tales como antologías o compilaciones y bases de datos de toda clase, que por la selección o disposición de las materias constituyan creaciones intelectuales, sin perjuicio de los derechos de autor que subsistan sobre los materiales o datos;

Proyectos, planos, maquetas y diseños de obras arquitectónicas y de ingeniería;

Ilustraciones, gráficos, mapas y diseños relativos a la geografía, la topografía, y en general a la ciencia;

Programas de ordenador; y,

Adaptaciones, traducciones, arreglos, revisiones, actualizaciones y anotaciones; compendios, resúmenes y extractos; y, otras transformaciones de una obra, realizadas con expresa autorización de los autores de las obras originales, y sin perjuicio de sus derechos.

SECCIÓN V

DISPOSICIONES ESPECIALES SOBRE CIERTAS OBRAS

PARÁGRAFO PRIMERO

DE LOS PROGRAMAS DE ORDENADOR

Art. 28. Los programas de ordenador se consideran obras literarias y se protegen como tales. Dicha protección se otorga independientemente de que hayan sido incorporados en un ordenador y cualquiera sea la forma en que estén expresados, ya sea en forma legible por el hombre (código fuente) o en forma legible por máquina (código objeto), ya sean programas operativos y programas aplicativos, incluyendo diagramas de flujo, planos, manuales de uso, y en general, aquellos elementos que conformen la estructura, secuencia y organización del programa.



Art. 29. Es titular de un programa de ordenador, el productor, esto es la persona natural o jurídica que toma la iniciativa y responsabilidad de la realización de la obra. Se considerará titular, salvo prueba en contrario, a la persona cuyo nombre conste en la obra o sus copias de la forma usual.

Dicho titular está además legitimado para ejercer en nombre propio los derechos morales sobre la obra, incluyendo la facultad para decidir sobre su divulgación.

El productor tendrá el derecho exclusivo de realizar, autorizar o prohibir la realización de modificaciones o versiones sucesivas del programa, y de programas derivados del mismo.

Las disposiciones del presente artículo podrán ser modificadas mediante acuerdo entre los autores y el productor.

Art. 30. La adquisición de un ejemplar de un programa de ordenador que haya circulado lícitamente, autoriza a su propietario a realizar exclusivamente:

Una copia de la versión del programa legible por máquina (código objeto) con fines de seguridad o resguardo;

Fijar el programa en la memoria interna del aparato, ya sea que dicha fijación desaparezca o no al apagarlo, con el único fin y en la medida necesaria para utilizar el programa; y,

Salvo prohibición expresa, adaptar el programa para su exclusivo uso personal, siempre que se limite al uso normal previsto en la licencia. El adquirente no podrá transferir a ningún título el soporte que contenga el programa así adaptado, ni podrá utilizarlo de ninguna otra forma sin autorización expresa, según las reglas generales.

Se requerirá de autorización del titular de los derechos para cualquier otra utilización, inclusive la reproducción para fines de uso personal o el aprovechamiento del programa por varias personas, a través de redes u otros sistemas análogos, conocidos o por conocerse.



Art. 31. No se considerará que exista arrendamiento de un programa de ordenador cuando éste no sea el objeto esencial de dicho contrato. Se considerará que el programa es el objeto esencial cuando la funcionalidad del objeto materia del contrato, dependa directamente del programa de ordenador suministrado con dicho objeto; como cuando se arrienda un ordenador con programas de ordenador instalados previamente.

Art. 32. Las excepciones al derecho de autor establecidas en los artículos 30 y 31 son las únicas aplicables respecto a los programas de ordenador.

Las normas contenidas en el presente Parágrafo se interpretarán de manera que su aplicación no perjudique la normal explotación de la obra o los intereses legítimos del titular de los derechos.

LEY DE EDUCACIÓN

LEY Nº. 127 DEL 15 DE ABRIL DE 1983

LA CÁMARA NACIONAL DE REPRESENTANTES

Considerando:

La necesidad de armonizar las normas sobre educación con los principios constantes en la Constitución Política; La utilidad de recoger las experiencias y el resultado del amplio debate en materia educativa y realizado entre los funcionarios del Ministerio de Educación y los representantes de los distintos sectores que conforman el área educativa; que es necesario, de conformidad con la estructura actual del Ministerio de Educación, dictar una ley específica consagrada únicamente a la educación, por las particularidades que tiene este sector; que es preciso establecer un marco claro y mecanismos ágiles para el desenvolvimiento de la educación nacional; y, En el ejercicio de la facultad contemplada en el artículo 66 de la Constitución

EXPIDE LA SIGUIENTE LEY DE EDUCACIÓN

Título Primero

PRINCIPIOS GENERALES

Capítulo I



OBJETO DE LA LEY

Art. 1.- La presente Ley tiene como objeto fijar los principios y fines generales que deben inspirar y orientar la educación, establecer las regulaciones básicas para el gobierno, organización y más funciones del sistema educativo y determinar las normas fundamentales que contribuyen a promover y coordinar el desarrollo integral de la educación.

Capítulo II

PRINCIPIOS Y FINES

Art. 2.- La educación se rige por los siguientes principios:

- a) La educación es deber primordial del Estado, que lo cumple a través del Ministerio de Educación y de las Universidades y Escuelas Politécnicas del país;
 - b) Todos los ecuatorianos tienen el derecho a la educación integral y la obligación de participar activamente en el proceso educativo nacional;
 - c) Es deber y derecho primario de los padres, o de quienes los representan, dar a sus hijos la educación que estimen conveniente. El Estado vigilará el cumplimiento de este deber y facilitará el ejercicio de este derecho;
 - d) El Estado garantiza la libertad de enseñanza de conformidad con la Ley;
 - e) La educación oficial es laica y gratuita en todos sus niveles. El Estado garantiza la educación particular;
 - f) La educación tiene sentido moral, histórico y social; se inspira en los principios de nacionalidad, democracia, justicia social, paz, defensa de los derechos humanos y está abierta a todas las corrientes del pensamiento universal;
 - g) El Estado garantiza la igualdad de acceso a la educación y la erradicación del analfabetismo;
 - h) La educación se rige por los principios de unidad, continuidad, secuencia, flexibilidad y permanencia;
 - i) La educación tendrá una orientación democrática, humanística, investigativa, científica y técnica, acorde con las necesidades del país; y,
1. La educación promoverá una auténtica cultura nacional, esto es, enraizada en la realidad del pueblo ecuatoriano.



Art. 3.- Son fines de la educación ecuatoriana:

- a) Preservar y fortalecer los valores propios del pueblo ecuatoriano, su identidad cultural y autenticidad dentro del ámbito latinoamericano y mundial;
 - b) Desarrollar la capacidad física, intelectual, creadora y crítica del estudiante, respetando su identidad personal para que contribuya activamente a la transformación moral, política, social, cultural y económica del país;
 - c) Propiciar el cabal conocimiento de la realidad nacional para lograr la integración social, cultural y económica del pueblo y superar el subdesarrollo en todos sus aspectos;
 - d) Procurar el conocimiento, la defensa y el aprovechamiento óptimo de todos los recursos del país;
 - e) Estimular el espíritu de investigación, la actividad creadora y responsable en el trabajo, el principio de solidaridad humana y el sentido de cooperación social;
 - f) Atender preferentemente la educación preescolar, escolar, la alfabetización y la promoción social, cívica, económica y cultural de los sectores marginados; y,
- Para cumplir a cabalidad con los fines de la educación, el Ministerio promoverá la participación activa y dinámica de las instituciones públicas y privadas y de la comunidad en general.

Título Segundo

ESTRUCTURA DEL SISTEMA EDUCATIVO

Capítulo I

ESTRUCTURA GENERAL

Art. 4.- (Reformado por el Art. 1 de la Ley 150, R.O. 918, 20-IV-92).- El sistema educativo nacional comprende dos subsistemas:

- a) El escolarizado; y,
- b) El no escolarizado.

En el sistema educativo nacional se garantiza la Educación Intercultural Bilingüe que, asimismo, comprenderá dos subsistemas:

- a) El escolarizado; y,
- b) El no escolarizado.



Art. 5.- El subsistema escolarizado se emplea en los establecimientos determinados por la Ley y comprende:

- a) La educación regular que se somete a las disposiciones reglamentarias sobre el límite de edad, secuencia de niveles y duración de cursos;
- b) La educación compensatoria, que tiene un régimen especial y se la ofrece a quienes no ingresan a los niveles de educación regular o no los concluyen; y,
- c) La educación especial, destinada a estudiantes excepcionales por razones de orden físico, intelectual, psicológico o social.

Art. 6.- El subsistema no escolarizado procura el mejoramiento educacional, cultural y profesional a través de programas especiales de enseñanza-aprendizaje y difusión, mediante los esfuerzos e iniciativas públicos o privados.

Art. 7.- La educación regular comprende los siguientes niveles:

- a) Pre primario;
- b) Primario; y, 1. Medio.

Art. 8.- La educación en el nivel pre primario tiende al desarrollo del niño y sus valores en los aspectos motriz, biológico, psicológico, ético y social, así como a su integración a la sociedad con la participación de la familia y el Estado.

Art. 10.- La educación en el nivel medio comprende los tres ciclos: básico, diversificado y especializado.

Art. 11.- El ciclo básico inicia la formación del nivel medio en el que se promueve una cultura general básica y se desarrollan actividades de orientación, que permiten al estudiante seleccionar la especialidad en el ciclo diversificado y habilitarle para el trabajo.

Art. 12.- El ciclo diversificado procura la preparación interdisciplinaria que permita la integración del alumno a las diversas manifestaciones del trabajo y la continuación de los estudios en el ciclo post-bachillerato o en el nivel superior, atendiendo a los requerimientos del desarrollo social y económico del país y a las diferencias y aspiraciones individuales.



Las diversas modalidades se organizan de acuerdo con las necesidades del desarrollo científico, económico y cultural del país y aseguran, con sentido integral, la formación humanística y técnica.

Título Tercero

ORGANIZACIÓN Y ADMINISTRACIÓN DEL SISTEMA EDUCATIVO

Capítulo I

EL MINISTERIO DE EDUCACIÓN

Art. 23.- El Ministerio de Educación y Cultura es responsable del funcionamiento del sistema educativo nacional, de la formulación y ejecución de la política cultural y deportiva y de la difusión del desarrollo científico y tecnológico.

Capítulo III

ESTRUCTURA DEL MINISTERIO

Art. 28.- (Agregado el inc. 2o. por el Art. 2 de la Ley 150, R.O. 918, 20-IV-92).- El Ministerio de Educación cuenta para su funcionamiento, además de las Subsecretarías, con las direcciones nacionales especializadas y las oficinas técnicas que se determinen en el Reglamento, de acuerdo con los requerimientos del desarrollo educativo del país. Además contará con una organización integrada por unidades de asesoramiento, de planificación y de ejecución.

Capítulo V

DE LA ORIENTACIÓN Y BIENESTAR ESTUDIANTILES

Art. 37.- El Ministerio de Educación planificará y ejecutará obligatoriamente la orientación dentro de un criterio integral, considerándola consubstancial al proceso educativo y como una acción interdisciplinaria con responsabilidad participativa de la comunidad educativa, conducida por profesionales especializados y desarrollada al interior del currículum.

Los servicios de bienestar estudiantil serán aplicados de igual manera con criterio especializado para facilitar un óptimo aprovechamiento de los recursos humanos, económicos y materiales en el proceso educativo, dentro de un contexto social.



Capítulo VII

INFRAESTRUCTURA EDUCATIVA

Título Cuarto

EL RÉGIMEN ESCOLAR

Capítulo I

EL AÑO LECTIVO, EL INGRESO Y LA EVALUACIÓN

Art. 45.- El año lectivo comprenderá ciento ochenta y cinco días laborables, incluidos los períodos de exámenes. Los calendarios, períodos y días de descanso para los regímenes de Sierra y Oriente, Costa y Galápagos, serán establecidos por el Ministerio de Educación.

Art. 46.- El Ministerio de Educación posibilitará la atención a la población escolar que requiera del cambio de régimen escolar evitando la pérdida de tiempo. Para el efecto tenderá a crear, al menos en todas las capitales de provincias, establecimientos, tanto de nivel primario como de nivel medio, que funcionen según el calendario de los establecimientos del otro régimen escolar.

Art. 47.- Los planteles educativos podrán adoptar el sistema de una o dos jornadas de trabajo diario, previa la justificación de su conveniencia y la autorización del Ministro.

En cualquier tipo de jornada, los establecimientos educativos se sujetarán a los períodos de clase determinados en el Reglamento.

Art. 48.- La admisión de alumnos en los diversos niveles y modalidades del sistema educativo se regirá por esta Ley y el Reglamento.

Art. 49.- La evaluación en todos los niveles y etapas de la educación será permanente e integral.

Art. 50.- Los establecimientos educativos nocturnos estarán sujetos a un régimen adecuado para el cumplimiento de su finalidad, adoptando las regulaciones necesarias para que estos planteles cumplan con los objetivos y fines educativos contemplados en la Ley y su Reglamento.

Título Quinto



PERSONAL DOCENTE Y ADMINISTRATIVO

Capítulo I

EL MAGISTERIO NACIONAL

Art. 52.- El Magisterio Nacional está formado por los profesionales de la educación y por aquellos que cumplan labores docentes o que desempeñen funciones técnico-administrativas especializadas en el sistema educativo.

Art. 53.- Quienes poseen título de profesionales de la educación tienen derecho prioritario para ser nombrados en funciones del ramo educativo.



CAPÍTULO III

METODOLOGÍA DE INVESTIGACIÓN

3.1. Tipos De Investigación

Entre los diversos tipos de investigación que existen, para el desarrollo del presente proyecto se utilizaran para:

Conocer las necesidades actuales de las necesidades actuales del colegio.

Recopilar la información necesaria para sustentar el desarrollo del sistema.

3.1.1. Investigación Histórica

Este tipo de investigación me ayudo hacer una investigación al pasado para analizar los procesos generales del sistema de control académico, tomando en cuenta cuanto lleva a la secretaria el realizar los diferentes procesos de gestión académica.

Luego de analizar cómo se llevaba los diferentes procesos en años anteriores, y como se pretende automatizar con el fin de mejorar, para esto se definirá por diferentes módulos los cuales serán viables, contemplando las funcionalidades que prestará al usuario dentro del contexto que implica el uso del Sistema Automatizado el cual consta de los siguientes módulos: Mantenimiento, Seguridad y Transaccional.

3.1.2. Investigación Bibliográfica Documental

Este tipo de investigación permitió recopilar y analizar la información la cual estaba en los documentos, carpetas que tenía el colegio para dar una posible solución a los problemas que tenía.



Este tipo de investigación es de mucha importancia ya que nos ayudara en el análisis de la información contenida en catálogos, revistas, manuales técnicos, proyectos que tengan información relacionada con la Implementación de un sistema de control académico.

Cabe recalcar que los documentos antes mencionados nos sirven solamente como guía, ya que el concepto y contenido deberán ser analizados y examinados cuidadosamente, a fin de poder obtener una estructuración lógica y ordenada de ideas y conceptos que permitan desglosar un documento por apertura a todo tipo de lector.

Es lógico suponer que estos documentos antes mencionados mantienen su identidad de autor por lo cual se mantendrá el respeto y sus derechos, de esta manera toda la información que rescatemos como ayuda mantendrá un cruce de la misma con otros textos y autores que manejan la misma temática. En lo referente a la información que pueda obtener del internet será tratada con las mismas consideraciones advertidas anteriormente, por consiguiente la estructuración lógica de ideas que generen el documento para el proyecto serán de exclusiva responsabilidad del investigador, amalgamando ideas lógicas y técnicamente estructuradas.

Lo que nos permitirá conocer, interpretar, comparar y enfocar criterios, opiniones conceptualizaciones que las mismas nos conducirán a determinar conclusiones y recomendaciones del proyecto planteado.

3.1.3. Investigación Descriptiva

Esta investigación me permite iniciar el primer punto en el desarrollo de mi proyecto que es el planteamiento del problema ya que mediante su estructura me permitió buscar explicaciones razonables a la problemática existente determinando las causas que generan las mismas y los efectos que ocasionaron, es lógico suponer que toda esta información se realizó con problemas reales y objetivos pudiendo tomar datos reales y concretos para el análisis y síntesis de la



información, asimismo en determinar la estructura lógica del problema de mi tema “Sistema de Control Académico”.

3.1.4. Investigación Explicativa

Esta investigación me permite saber las causas y problemas que tiene el colegio en los procesos de gestión académica, lo cual se debe a que la institución no cuenta con la automatización de estos, razón por la cual se procederá a la automatización de los mismos logrando así la continuidad del desarrollo del sistema mencionado.

3.2. Métodos Empleados en la Investigación

3.2.1. Método Inductivo

Por medio del método inductivo se podrá tener una perspectiva para ver las cosas más simples considerando los casos particulares, para en base a eso, establecer casos generales es decir, analizar el tiempo que la secretaria emplea en los diferentes procesos, sea este ingreso o búsqueda, y establecer una mejor visión para los principales impactos generados con la aplicación del proyecto.

3.2.2. Método Deductivo

Este método me ayudo para hacer en mi sistema consulta, reportes tanto de manera general que pueda ver el usuario todas las calificaciones que tiene el alumno así como una calificación en especial que quiere ver o saber el alumno en cualquiera de la materia.

Tomando en consideración casos generales, se buscará la forma de obtener detalles específicos para atacar cada uno de los problemas implícitos en el proceso.

Cada uno de los procesos tienen sub procesos los cuales se manejan dentro de un mismo, la entrega de documentos por parte de los alumnos crean el manejo de las Inscripciones lo que permite crear un nuevo proceso llamado Matriculación el cual constan los alumnos que cumplen con los documentos requeridos. El manejo del control de notas se lo obtiene de la información entregada por el docente el cual califica de la siguiente manera exámenes, actuaciones en clase, pruebas, deberes,



cuadernos. El control de docentes se lo maneja con una hoja la cual firman al momento de ingresar a la institución. Lo cual genera procesos de manera muy particular y para poder atacar cada uno de estos problemas se creara un módulo donde se pueda manipular toda esta información de una manera más eficaz.

El proyecto pretende cubrir en toda su totalidad cada una de las siguientes característica, ya sean estos realizados a través de menús, módulos, pantallas, reportes o procesos internos para el manejo de información, por este motivo separaremos en dos puntos importantes lo que este incluye así como lo que no se encuentra considerado en el mismo.

3.2.3. Método Histórico Lógico

Este método permite realizar paso a paso el desarrollo de un sistema informático ya que mediante este se puede tener su aplicación, inicialmente en el estudio y análisis de los procesos y procedimientos necesarios a automatizar en mi tema “Sistema de Control Académico” que posteriormente luego de haber terminado las tareas automatizables me permitió también verificar la recopilación de requerimientos que unidos a los procesos permitieron modelar las reglas del negocio hasta llegar a obtener un sistema de gestión de base de datos.

Por otro lado este método al permitirnos seguir una trayectoria definida, permitió mantener un orden cronológico estándar ya definido en el área de sistemas y aplicados al desarrollo del presente proyecto.

3.2.4. Método Analítico Sintético

Este método es de mucha importancia en la aplicación del proyecto ya que en el mismo se pudo realizar el análisis de la información teórica concerniente al marco referencial; al obtener gran cantidad de información referente al tema de proyecto y su aplicación, este método permitió ir determinando la información estrictamente necesaria como marco teórico y que se está utilizando en el sistema a desarrollarse por otro lado al realizar una síntesis de los procesos y procedimientos y tareas automatizables permitieron estructurar un diseño lógico del negocio preciso y aplicado a las necesidades reales del colegio.



Cabe recalcar que realizar el análisis y la síntesis de la información son pasos que los estamos ejecutando en todo momento en el desarrollo de proyecto, asimismo la estructuración tecnológica en el manejo y desarrollo de los códigos fuentes.

3.2.5. Método Sistémico

Este método me permite seguir de una forma ordenada el desarrollo de presente proyecto, y seguir paso a paso la realización de los capítulos anteriores y también el desarrollo del sistema de control académico.

3.3. Técnicas De Recolección De Información

3.3.1. Observación

La observación será una de las principales técnicas de recolección que se empleara para este proyecto, lo cual servirá para analizar el proceso en una visión muy general, y a momentos también enfocarse en detalles específicos que el autor considera necesarios para la misma.

La observación que realizaremos es una combinación de la observación directa e indirecta ya que estaremos en contacto con el problema o situación a mejorar, será participante porque nos adentraremos para conocer y tratar desde cerca con el objeto investigado, nos trasladaremos al lugar objeto de estudio.

Cuando realicemos la observación utilizaremos una ficha de observación para así poder obtener datos que nos ayuden en la confección del análisis y mejoramiento del control de gestión académica.

Con la observación de estos procesos podemos automatizar y desarrollar un software para mejorar la actividad de las secretarias.

3.3.2. Entrevista

La entrevista, porque es una técnica orientada a establecer un contacto directo con las personas consideradas fuentes de información directa.



Esta herramienta consiste básicamente en reunirse dos o varias personas y cuestionarlas en forma adecuada para obtener información.

Utilizaremos esta técnica para obtener datos consistentes mediante el diálogo o conversación, los cuales nos ayudarán al mejor manejo de la información veraz y concisa que nos proporcionarían el personal administrativo del colegio, ya que se establecerá una conversación mutua con el personal.

Pregunta No. 1

¿El colegio tiene algún sistema automatizado que controle los distintos procesos que tiene la institución con respecto a la gestión académica?

Respuesta: No

Análisis: Con esta pregunta se sabe que el colegio no tiene automatizado los procesos de gestión académica.

Pregunta No. 2

¿Estaría de acuerdo en implementar un sistema que automatice dichos procesos?

Respuesta: Si

Análisis: Con esta pregunta se sabe que se requiere implementar un sistema que logre automatizar los procesos de la institución.

Pregunta No. 3

¿Cree usted que con la implementación de un sistema automatizado mejore la atención a los usuarios al momento de inscripciones y matriculas?

Respuesta: Claro que sí, ya que la atención sería mucho más rápida tomando en cuenta que ahora sería automatizado, ya que se optimizaría tiempo.

Análisis: Con esta pregunta se desea saber si mejorará la atención lo cual se esta muy seguro, esto se lo puede observar al momento que se implemente el sistema en el colegio y se realice los distintos procesos.

Pregunta No. 4

¿Cree usted que se optimizará recursos con la implementación del sistema automatizado?



Respuesta: Si, ya que se ahorraría papel, tinta de impresión, carpetas y otros recursos más.

Análisis: Con esta pregunta se sabe que el colegio optimizará gastos, ya que no tendrá que consumir muchos recursos, debido a que la información se la guardará en una base de datos, y ya no se la almacenará físicamente en carpetas como se lo viene haciendo.

3.3.3. Encuesta

Para la realización de esta técnica de obtención de información hemos tomado en cuenta que es una población y que es una muestra ya que nos manejaremos con estos términos los cuales detallamos a continuación:

Población: Consiste en un conjunto de elementos, seres o eventos, concordantes entre sí en cuanto a una serie de características, de distinta clasificación de los cuales se desea obtener alguna información.

Para realizar un buen análisis debemos buscar información en todas aquellas personas que nos la pueden brindar es por eso que tenemos que utilizar instrumentos que nos permitan saber lo que piensan sobre el control de gestión académica.

Nuestra población es de 750 personas entre alumnos y docentes, es con esta que calcularemos la muestra que debemos aplicar para recopilar información valiosa la cual sea posible de tabular.

Muestra: Es un subconjunto, extraído de la población, (mediante técnicas de muestreo) cuyo estudio sirve para inferir características de toda la población.

La fórmula utilizada para calcular la muestra es:

$$n = \frac{N \times P \times Q \times Z^2}{(N-1)E^2 + P \times Q \times Z^2}$$

Significado de las Variables utilizadas en la fórmula.



n= tamaño muestra

N = población o universo

E = margen de error

P = probabilidad de éxito

Q = probabilidad de fracaso

Z = grados de confianza

Para la aplicación de la fórmula establecemos los siguientes valores:

n = Tamaño de la muestra

N = Población estudiantil 750

E = 5 % (0,05)

p = 50% (0,5)

q = 10 % (0,1)

z² = 1,56

$$n = \frac{(750)(0,5)(0,1)(1,56)}{(750-1)(0,05^2) + (0,5)(0,1)(1,56)} = 30$$

La población de la investigación está compuesta por un total de setecientos cincuenta personas, quienes de forma directa están relacionadas con la actividad del colegio. Razón por la que se realizó un censo, y no un muestreo de la población.

La población está compuesta por:



CURSOS	NUMERO DE PERSONAS
3 Primeros	100
3 Segundos	98
3 Terceros	95
4 Cuartos	145
4 Quitos	138
4 Sextos	133
Docentes	41
Total Población	750

Tabla Nº 2.
Población del Colegio
Fuente: Colegio Benjamín Carrión

Ver Anexo Nº1: Formato de Encuesta

Pregunta No. 1

¿Sabe si el colegio posee controles automatizados para el manejo de inscripciones, matrículas y notas?

RESPUESTA	SI	NO
Valores obtenidos	5	25
Porcentaje	17	83

Tabla Nº 3.
Resultados Pregunta 1
Fuente: Cuestionario de encuesta

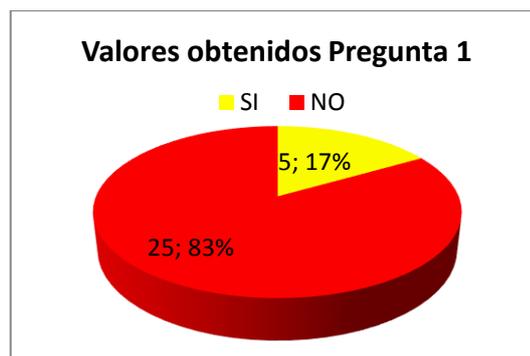


Gráfico Nº 10.
Resultados Pregunta 1
Fuente: Cuestionario de encuesta

Análisis:



Con esta pregunta se pretende saber si el colegio tiene o no un sistema de control académico, de acuerdo a las respuesta se ha observado que el 17% a respondido que si lo tiene y el 83% dice que no lo tiene, esto quiere decir que el colegio no tiene sistemas automatizados que faciliten el desarrollo de sus tareas.

Pregunta No. 2

¿Cree usted que la atención a los padres de familia y alumnado mejorará con la implantación de un sistema de control académico?

RESPUESTA	SI	NO
Valores obtenidos	25	5
Porcentaje	83	17

Tabla N° 4.
Resultados Pregunta 2
Fuente: Cuestionario de encuesta



Gráfico N° 11.
Resultados Pregunta 2
Fuente: Cuestionario de encuesta

Análisis:

Con esta pregunta se pretende saber si la atención que brinda el colegio podría ser más rápida y eficaz con la implementación de un sistema automatizado, para lo cual el 83% de las personas encuestadas a respondido que si y el 17% no, observamos de acuerdo al porcentaje de repuesta que la atención si mejorara.



Pregunta No. 3

¿Conoce usted el sistema de evaluación que se utiliza en el colegio?

RESPUESTA	SI	NO
Valores obtenidos	17	13
Porcentaje	57	43

Tabla Nº 5.
Resultados Pregunta 3
Fuente: Cuestionario de encuesta



Gráfico Nº 12.
Resultados Pregunta 3
Fuente: Cuestionario de encuesta

Análisis:

De acuerdo a la respuesta dada por los encuestados, el 57% a mencionado que si conoce el sistema de evaluación que utiliza el colegio y el 43% a dicho que no, se puede observar que no hay mucha diferencia en el valor de porcentaje.

Pregunta No. 4

¿Considera usted que el sistema de evaluación del colegio es bueno?

RESPUESTA	SI	NO
Valores obtenidos	19	11
Porcentaje	63	37

Tabla Nº 6.
Resultados Pregunta 4
Fuente: Cuestionario de encuesta

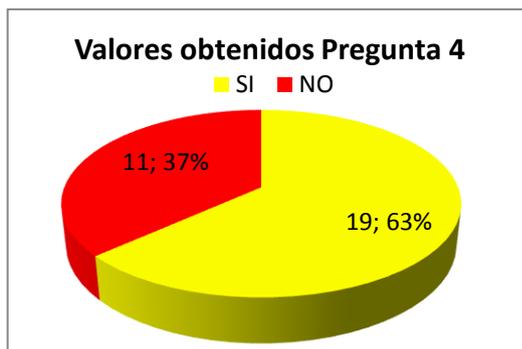


Gráfico N° 13.
Resultados Pregunta 4
Fuente: Cuestionario de encuesta

Análisis:

Con esta pregunta se trata de conocer si el sistema de evaluación del colegio es bueno, para lo cual el 63% ha votado por la respuesta si y el 37% por el no.

Pregunta No. 5

¿Al momento de las inscripciones y matriculas la atención es buena?

RESPUESTA	SI	NO
Valores obtenidos	16	14
Porcentaje	53	47

Tabla N° 7.
Resultados Pregunta 5
Fuente: Cuestionario de encuesta



Gráfico N° 14.
Resultados Pregunta 5
Fuente: Cuestionario de encuesta



Análisis:

Con esta pregunta se trata de saber si la atención al momento de inscripciones y matriculas es buena o no, de acuerdo a la encuesta realizada el 53% ha mencionado que si es buena y el 47% a mencionada que no lo es.

Pregunta No. 6

¿Con la creación del sistema académico se facilitara el trabajo de la secretaria?

RESPUESTA	SI	NO
Valores obtenidos	23	7
Porcentaje	77	23

Tabla N° 8.
Resultados Pregunta 6
Fuente: Cuestionario de encuesta

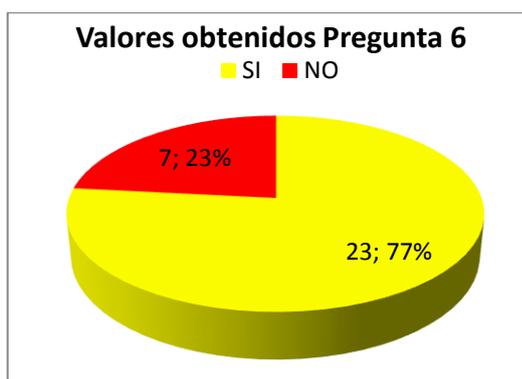


Gráfico N° 15.
Resultados Pregunta 6
Fuente: Cuestionario de encuesta

Análisis:

Con esta pregunta se pretende saber si el trabajo de la secretaria se le facilitaría y sería mejor, de acuerdo a la encuesta realiza el 77% a mencionado que el trabajo de la secretaria si se le facilitaría y sería mejor y el 23% a dicho que no se le facilitaría.



CAPITULO IV

DESARROLLO DE LA PROPUESTA

4.1. Diagnostico Situacional

El Colegio Dr. Manuel Benjamín Carrión Mora actualmente está en funcionamiento desde 1977 brindando a la comunidad del Comité del Pueblo, una educación de mejora continua dentro de estas mejoras esta la adquisición de un software para poder controlar los procesos de gestión académica, además cuenta con diferentes especialidades como: Contabilidad, Mecánica Industrial, Electromecánica y Químico Biólogo.

Realizado el análisis respectivo de la situación del Colegio, procedemos a puntualizar lo siguiente.

Actualmente la institución ha venido desarrollando sus actividades de una manera un tanto empírica se podría decir, ya que el modo como se conduce a la información generada es muy precaria, sin ninguna norma técnica, y sufre de una estructura muy vulnerable ya que los archivos son almacenados en hojas impresas.

Como se mencionó anteriormente el colegio está en proceso de adquisición de un sistema que le permita gestionar los procesos académicos, dicho software se lo está desarrollando y para esto el colegio cuenta con el hardware apropiado por lo que no se necesita hacer ninguna adquisición.



En cuanto a software cuenta con dos sistemas operativos Windows XP y 7, de igual manera tiene el paquete de Microsoft Office 2007 y 2010 y un antivirus el cual es NOD 32 todos estos sin licencia.

Cabe mencionar que para la realización del sistema se utilizará Visual Studio 2010 como IDE de desarrollo y SQLServer como gestor de base de datos.

En nuestro estudio realizado estamos proponiendo la realización de un sistema que si bien por el tiempo no se culminaran todos sus módulos, pero si abarcarán los principales, y estos a su vez solucionarán en gran medida las necesidades de automatización de los procesos de gestión académica.

4.2. Estructura Organizacional

Un organigrama estructural es la expresión gráfica de la estructura organizativa de la institución educativa en él se describen los diferentes niveles de los órganos de autoridad, coordinación y apoyo, así como sus interrelaciones, estos organigramas son muy importantes ya que ayuda a organizar un empresa o institución. En la actualidad se trata de diseñar orgánicos estructurales horizontales, o en espiral, es la institución la que de acuerdo a su modelo de gestión lo diseña.

La finalidad de una estructura organizacional es establecer un sistema de papeles que han de desarrollar los miembros de una entidad para trabajar juntos de forma óptima y que se alcancen las metas fijadas en la planificación.

Elementos de la organización: (requerimientos)

- Los objetivos deben ser verificables, precisos y realizables. Para que sean precisos deben ser cuantitativos y para ser verificables deben ser cualitativos.
- Tiene que haber una clara definición de los deberes, derechos y actividad de cada persona.



- Se tiene que fijar el área de autoridad de cada persona, lo que cada uno debe hacer para alcanzar las metas.
- Sabe cómo y dónde obtener la información necesaria para cada actividad. Cada persona debe saber dónde conseguir la información y le debe ser facilitada.

¿Qué es organizar?

1. Identificar y clasificar las actividades que se tienen que realizar en la empresa.
2. Agrupamos estas actividades.
3. A cada grupo de actividades le asignamos un director con autoridad para supervisar y tomar decisiones.
4. Coordinamos vertical y horizontalmente la estructura resultante.
5. Estructura organizacional

Ver Anexo N°2: Diagrama Organizacional Del Colegio “Dr. M.B.C.M”.

4.2.1. Orgánico Estructural

La estructura organizacional se refiere a la forma en que se dividen, agrupan y coordinan las actividades de la organización en cuanto a las relaciones entre los gerentes y los empleados, entre gerentes y gerentes y entre empleados y empleados. Los departamentos de la organización están estructurados formalmente por la función que realizan como se aprecia en el siguiente organigrama.

Ver Anexo N°3: Diagrama Orgánico Funcional del Colegio “Dr. M.B.C.M”.

4.2.2. Orgánico Funcional

La organización por funciones reúne, en un departamento, a todos los que se dedican a una actividad o a varias relacionadas, que se llaman funciones. Por ejemplo, una organización dividida por funciones puede tener departamentos para producción, mercadotecnia y ventas. El gerente de ventas de dicha



organización sería el responsable de la venta de todos los productos manufacturados por la empresa.

Es probable que la organización funcional sea la forma más lógica y básica de departamentalización, como se detalla a continuación:

Estructura Orgánico Funcional

Rector.- Es la persona encargada de controlar que todas las funciones de la Unidad Educativa estén funcionando de manera perfecta.

Vicerrector.- Se encarga de controlar las diferentes áreas que están a su cargo y rendir informes al rector.

Inspector General.- Se encarga de controlar a los alumnos y docentes que estén cumpliendo sus respectivas actividades.

Inspector de Disciplina.- Es la persona que se encarga de controlar que los Alumnos se encuentren debidamente uniformados y cumplan con las reglas de la institución para no tener ningún problema dentro de la misma.

Consejo Estudiantil.- Es el ente que representa a todos los estudiantes el cual se encarga de buscar las necesidades de los estudiantes y tramitar las debidas soluciones con las autoridades correspondientes para llegar a acuerdos entre los alumnos y autoridades de la Institución.

Consejo de Profesores.- Es el ente que representa a todos los profesores dentro de la Institución el cual hace cumplir a cabalidad con todos los derechos de los profesores así como hacer cumplir las obligaciones que deben realizar los profesores.

Consejo de Padres de Familia.- ente encargado de representar a todos lo padres de familia de los alumnos del colegio, se encarga de organizar el comité central y las directivas de cursos.

Secretario(a) General.- Se encarga del manejo de Información de todos los alumnos, verifica que el trabajo de los secretarios(as) estén realizados de manera correcta, también es la persona quien se encarga de todo los trámites correspondientes que necesite realizar la institución.



Secretario(a).-es la persona quien ayuda al secretario(a) general en cumplir sus funciones, la cual cumple varias funciones como atender a los Alumnos, Profesores y Padres de familia de manera permanente es quien recibe toda la información para clasificarla y entregarla a la persona correspondiente.

BOBE.- departamento encargado de dar orientación vocacional tanto a los alumnos como a los padres de familia.

Colecturía.- departamento encargado del aspecto financiero de la institución.

Guardalmacén.- persona encargada de cuidar el almacén donde se encuentran los uniformes de los alumnos y profesores.

Servicios Generales.- departamento encargado de brindar servicios de biblioteca, talleres, laboratorios, bar, etc.

Auxiliar de Servicios.- es la persona encargada de mensajería, conserjería y guardianía.

4.3. Infraestructura Informática

El Colegio Benjamín Carrión cuenta con la siguiente tecnología para el correcto funcionamiento del sistema de control académico por lo que no se necesita hacer ningún cambio o adquirir nueva tecnología.

4.3.1. Hardware

El colegio tiene el siguiente hardware por cada departamento como se detalla a continuación:

INFRAESTRUCTURA DE HARDWARE		
Nº	Área	Detalle del Hardware
1	Rectorado	<ul style="list-style-type: none">➤ Procesador Intel Core 2 Quad de 2.4GHz.➤ Memoria RAM 1GB.➤ Disco duro de 160.➤ Monitor
		<ul style="list-style-type: none">➤ Procesador Intel Core 2 Duo de 1.86 GHz.



2	Vicerrectorado	<ul style="list-style-type: none">➤ Memoria RAM de 1GB.➤ Disco Duro 160Gb.➤ Monitor
3	Inspección	<ul style="list-style-type: none">➤ Procesador Intel Pentium IV de 1.4 GHz.➤ Disco Duro 80Gb➤ Memoria RAM de 256MB➤ Monitor
4	Secretaría	<ul style="list-style-type: none">➤ 2 Procesadores Intel Core 2 Duo de 2.2 GHz y Core 2 Quad 2.8 GHz.➤ 2 Disco Duro 500GB.➤ 2 Memoria RAM de 2GB.➤ 2 Monitor.➤ Impresora Hp a laser.
5	DOBE	<ul style="list-style-type: none">➤ 2 Procesadores Intel Pentium IV de 1.4 GHz y 1.5 GHz.➤ 2 Disco Duro 80Gb.➤ 2 Memorias RAM de 256MB y 512MB.➤ 2 Monitor.
6	Colecturía	<ul style="list-style-type: none">➤ 3 Procesadores Core 2 Duo de 2.2 GHz.➤ 3 Memorias RAM de 2 GB.➤ 3 Discos duros de 300 GB.➤ 3 Monitores.

Tabla N° 9.
Infraestructura de Hardware
Fuente: Colegio Dr. MBCM

4.3.2. Software

El colegio cuenta con el siguiente software el mismo que se detalla a continuación en la tabla:

INFRAESTRUCTURA DE SOFTWARE				
Nº	Área	Software	Versión	Licencia
		<ul style="list-style-type: none">➤ Microsoft Windows 7	Profesional	No



1	Rectorado	<ul style="list-style-type: none"> ➤ Microsoft Office ➤ NOD 32 	2007 2.5	No No
2	Vicerrectorado	<ul style="list-style-type: none"> ➤ Microsoft Windows 7 ➤ Microsoft Office ➤ NOD 32 	Profesional 2007 2.5	No No No
3	Inspección	<ul style="list-style-type: none"> ➤ Microsoft Windows XP ➤ Microsoft Office ➤ NOD 32 	Serví pack 2 2007 2.5	No No No
4	Secretaría	<ul style="list-style-type: none"> ➤ Microsoft Windows XP ➤ Microsoft Office ➤ NOD 32 	Serví pack 2 2007 2.5	No No No
5	DOBE	<ul style="list-style-type: none"> ➤ Microsoft Windows XP ➤ Microsoft Office ➤ NOD 32 ➤ Sistema de ficha de datos 	Serví pack 2 2007 2.5	No No No No
6	Colecturía	<ul style="list-style-type: none"> ➤ Microsoft Windows 7 ➤ Microsoft Office ➤ NOD 32 	Profesional 2010 2.5	No No No

Tabla N° 10.
Infraestructura de Software
Fuente: Colegio Dr. MBCM

4.3.3. Comunicaciones

La institución educativa cuenta con una red LAN (Red de Área Local) alámbrica y su ISP o Proveedor de Servicio de Internet es CNT, esta red está diseñada mediante una topología estrella, cabe recalcar que el internet llega mediante vía telefónica el cual está conectado a un modem y está a su vez a un router el cual está conectado a un hub y de este se distribuye los servicios de internet para los ordenadores, dentro de los activos y pasivos de red se encuentran los siguientes:



Activo

- Un Modem.
- Un Router.
- Un Hub.
- 10 Computadores.
- 2 Impresoras

Pasivos

- Cable UTP de categoría 5.
- Patch
- Jacks o conectores RJ-45

4.3.3.1. Topología en estrella:

Una red en estrella es una red en la cual las estaciones están conectadas directamente a un punto central y todas las comunicaciones se han de hacer necesariamente a través de éste. Los dispositivos no están directamente conectados entre sí, además de que no se permite tanto tráfico de información.

Dado su transmisión, una red en estrella activa tiene un nodo central *activo* que normalmente tiene los medios para prevenir problemas relacionados con el eco.

Se utiliza sobre todo para redes locales. La mayoría de las redes de área local que tienen un enrutador (router), un conmutador (switch) o un concentrador (hub) siguen esta topología. El nodo central en estas sería el enrutador, el conmutador o el concentrador, por el que pasan todos los paquetes.

Todos los elementos de la red se encuentran conectados directamente mediante un enlace punto a punto al nodo central de la red en este caso al hub, quien se encarga de gestionar las transmisiones de información por toda la red estrella. En este caso, todos los cables están conectados hacia un solo sitio.



Ventajas

- Si una PC se desconecta o se rompe el cable solo queda fuera de la red esa PC.
- Fácil de agregar, reconfigurar arquitectura PC.
- Fácil de prevenir daños o conflictos.
- Centralización de la red

Desventajas

- Si el nodo central falla, toda la red deja de transmitir.
- Es costosa, ya que requiere más cable que las topologías bus o anillo.
- El cable viaja por separado del concentrador a cada computadora.

Ver Anexo N°4: Diagrama de red del Colegio “Dr. M.B.C.M”.

4.3.4. Recurso Humano Técnico

Actualmente la institución cuenta con una sola persona para la administración de la red de comunicaciones y mantenimiento en general la cual se menciona a continuación:

- Víctor Manuel Rodríguez tiene 8 años de experiencia y posee un título de Tecnólogo.

4.4. Descripción De Alternativas

A continuación se detalla en la siguiente tabla las especificaciones técnicas que se tomaran en cuenta en el desarrollo del sistema.



ESPECIFICACIONES TÉCNICAS	
LEVANTAMIENTO DE REQUERIMIENTOS	
PLATAFORMA	TRES CAPAS
METODOLOGÍA	RUP
DISEÑO DEL SISTEMA	RATIONAL ROSE
MODELOS DE NEGOCIOS EN RATIONAL ROSE	FÍSICO
	LÓGICO
	SCRIPT DE BDD
FRONT END	C#
BACK END	SQL SERVER
STANDARES	DISEÑO
	PROGRAMACIÓN
	BASE DE DATOS
GARANTÍA TÉCNICA	6 MESES
SOPORTE TÉCNICO	6 MESES
METODOLOGÍA DE CAPACITACION	FACE TO FACE
PRUEBAS	UNIDAD
	INTEGRACIÓN
	VALIDACIÓN
	SEGURIDAD
TIEMPO	6 MESES
COSTO	\$ 3500

Tabla N° 11.
Especificaciones Técnicas
Fuente: Daniel Granizo

4.4.1. Alternativa 1: Contratar A La Empresa Sidesoft Cia. Ltda Para El Desarrollo Del Software.

La presente alternativa es presentada por la empresa Sidesoft Cía. Ltda. desarrolladora de software en la misma que especifica el aspecto técnico, económico y las garantías.



PRESUPUESTO ALTERNATIVA 1: SIDESOFT CIA. LTDA			
ESPECIFICACIONES TÉCNICAS		CUMPLE	NO CUMPLE
LEVANTAMIENTO DE REQUERIMIENTOS		X	
PLATAFORMA	TRES CAPAS	X	
METODOLOGÍA	RUP	X	
DISEÑO DEL SISTEMA	RATIONAL ROSE		X
MODELOS DE NEGOCIOS EN RATIONAL ROSE	FÍSICO		X
	LÓGICO		X
	SCRIPT DE BDD		X
FRONT END	C#		X
BACK END	SQL SERVER	X	
STANDARES	DISEÑO	X	
	PROGRAMACIÓN	X	
	BASE DE DATOS	X	
GARANTÍA TÉCNICA	6 MESES		X
SOPORTE TÉCNICO	6 MESES	X	X
METODOLOGÍA DE CAPACITACION	FACE TO FACE	X	
PRUEBAS	UNIDAD	X	
	INTEGRACIÓN	X	
	VALIDACIÓN	X	
	SEGURIDAD	X	
TIEMPO	6 MESES	X	
COSTO	\$ 3500		

Tabla Nº 12.
Propuesta alternativa 1
Fuente: Empresa SIDESOFT CIA. LTDA

4.4.2. Alternativa 2: Contratar A La Empresa Logic Studio S.A Para El Desarrollo Del Software.

La presente alternativa es presentada por una empresa desarrolladora de software en la misma que especifica el aspecto técnico, económico y las garantías.



PRESUPUESTO ALTERNATIVA 2: LOGIC STUDIO S.A			
ESPECIFICACIONES TÉCNICAS		CUMPLE	NO CUMPLE
LEVANTAMIENTO DE REQUERIMIENTOS		X	
PLATAFORMA	TRES CAPAS	X	
METODOLOGÍA	RUP		X
DISEÑO DEL SISTEMA	RATIONAL ROSE	X	
MODELOS DE NEGOCIOS EN RATIONAL ROSE	FÍSICO	X	
	LÓGICO		X
	SCRIPT DE BDD	X	
FRONT END	C#	X	
BACK END	SQL SERVER	X	
STANDARES	DISEÑO		X
	PROGRAMACIÓN		X
	BASE DE DATOS		X
GARANTÍA TÉCNICA	6 MESES	X	
SOPORTE TÉCNICO	6 MESES	X	
METODOLOGÍA DE CAPACITACION	FACE TO FACE	X	
PRUEBAS	UNIDAD	X	
	INTEGRACIÓN	X	
	VALIDACIÓN	X	
	SEGURIDAD	X	
TIEMPO	6 MESES	X	
COSTO	\$ 4000		X

Tabla Nº 13.

Propuesta alternativa 1.

Fuente: Empresa LOGIC STUDIO S.A.

4.4.3. ***Alternativa 3: Proyecto De Tesis***

La presente alternativa es presentada por mi persona en el mismo específico el aspecto técnico, el aspecto económico, el tiempo de ejecución y las garantías.



PRESUPUESTO ALTERNATIVA 3: PROYECTO DE TESIS			
ESPECIFICACIONES TÉCNICAS		CUMPLE	NO CUMPLE
LEVANTAMIENTO DE REQUERIMIENTOS		X	
PLATAFORMA	TRES CAPAS	X	
METODOLOGÍA	RUP	X	
DISEÑO DEL SISTEMA	RATIONAL ROSE	X	
MODELOS DE NEGOCIOS EN RATIONAL ROSE	FÍSICO	X	
	LÓGICO	X	
	SCRIPT DE BDD	X	
FRONT END	C#	X	
BACK END	SQL SERVER	X	
STANDARES	DISEÑO	X	
	PROGRAMACIÓN	X	
	BASE DE DATOS	X	
GARANTÍA TÉCNICA	6 MESES	X	
SOPORTE TÉCNICO	6 MESES	X	
METODOLOGÍA DE CAPACITACION	FACE TO FACE	X	
PRUEBAS	UNIDAD	X	
	INTEGRACIÓN	X	
	VALIDACIÓN	X	
	SEGURIDAD	X	
TIEMPO	6 MESES	X	
COSTO	\$ 0	X	

Tabla Nº 14.
Propuesta Proyecto de Tesis.
Fuente: Daniel Granizo.

4.5. Evaluación Y Selección De Alternativas

Con la finalidad de poder evaluar las alternativas presentadas por las casas oferentes es necesario determinar factores de comparación y evaluación que me



permitan discernir entre cada uno de los elementos que componen las diferentes alternativas:

70% a lo técnico, 20% a lo económico, 5% al soporte técnico y 5% a la garantía técnica.

Para poder determinar un valor cuantitativo a las especificaciones técnicas se ha diseñado pesos para cada uno de ellos y de acuerdo a su importancia serán evaluadas las diferentes alternativas.

4.5.1. Evaluación Técnica

A continuación se muestra la evaluación técnica realizada a cada una de las alternativas, para ello hemos tomado en cuenta un porcentaje del 70% si la suma de las especificación técnicas es de 100, ya que a partir de este valor se procederá a evaluar cada alternativa.

EVALUACIÓN DE ESPECIFICACIONES TÉCNICAS					
ESPECIFICACIONES TÉCNICAS		PESO	ALTERNATIVA 1	ALTERNATIVA 2	ALTERNATIVA 3
PLATAFORMA	TRE S CAPAS	10	10	9	10
METODOLOGÍA	RUP	10	10	8	10
MODELOS DE NEGOCIOS	FÍSICO	10	10	10	10
	LÓGICO	10	8	10	10
	SCRIPT DE BDD	10	10	10	10
FRONT END	C#	10	10	8	10
BACK END	SQL SERVER	10	10	10	10
STANDARES	DISEÑO	10	8	10	10
	PROGRAMACIÓN	10	8	10	10
	BASE DE DATOS	10	8	10	10
TOTAL		100	92	95	100
PORCENTAJE		70%	64,4%	66,5%	70%

Tabla N° 15.
Evaluación de especificaciones técnicas.
Fuente: Daniel Granizo



4.5.2. Evaluación De Garantía Económica

A continuación se muestra la evaluación del soporte técnico de las distintas alternativas mencionadas, para la cual hemos tomado en cuenta un porcentaje del 20% si el costo es de cero a partir de este valor nos guiaremos para poder evaluar.

EVALUACIÓN GARANTÍA ECONÓMICA		
ECONOMICO	COSTO	PORCENTAJE
ALTERNATIVA 1	\$ 3.500	17,5%
ALTERNATIVA 2	\$ 4.000	0%
ALTERNATIVA 3	\$ 0	20%

Tabla N° 16.
Evaluación de garantía económica
Fuente: Daniel Granizo

4.5.3. Evaluación De Garantía Técnica

A continuación se muestra la evaluación de la garantía técnica de las distintas alternativas mencionadas, para poder evaluar hemos tomado en cuenta un porcentaje del 5% si la garantía técnica es de 6 meses, a partir de este valor se procederá a evaluar las alternativas.

EVALUACIÓN DE GARANTÍA TÉCNICA		
GARANTÍA TÉCNICA	TIEMPO	PORCENTAJE
alternativa 1	5 meses	4,16%
alternativa 2	6 meses	5%
alternativa 3	6 meses	5%

Tabla N° 17.
Evaluación de garantía técnica.
Fuente: Daniel Granizo

4.5.4. Evaluación De Soporte Técnico

A continuación se muestra la evaluación del soporte técnico de las distintas alternativas mencionadas, para lo cual hemos tomado en cuenta un porcentaje del 5% si el soporte técnico es de 6 visitas, a partir de este se procederá a evaluar las alternativas.



EVALUACIÓN SOPORTE TÉCNICO		
ALTERNATIVA	No DE VISITAS	PORCENTAJE
alternativa 1	6	5%
alternativa 2	3	2,50%
alternativa 3	6	5%

Tabla N° 18.
Evaluación soporte técnico
Fuente: Daniel Granizo

4.6. Factibilidad Técnica

Luego de realizar las respectivas evaluaciones a cada alternativa se ha observado que: La Alternativa 1 en especificaciones técnicas ha obtenido un puntaje de 92 lo que representa un 64,4%, en lo económico ha obtenido un porcentaje del 17,5%, en la garantía técnica ha obtenido un porcentaje de 4,16%, en el soporte técnico ha obtenido un 5%, siendo así un porcentaje total para esta alternativa del 91,06%.

La Alternativa 2 en especificaciones técnicas ha obtenido un puntaje de 95 lo que representa un porcentaje del 66,5%, en lo económico ha obtenido un porcentaje del 0% ya que el costo es de \$4000.00 que es el más alto, en la garantía técnica ha obtenido un porcentaje de 5%, en el soporte técnico ha obtenido un 5%, siendo así un porcentaje total para esta alternativa del 76,5%.

Finalmente la alternativa 3 que es el proyecto de tesis en especificaciones técnicas ha obtenido un puntaje de 100 ya cumple con todo lo necesario lo que representa un porcentaje del 70%, en lo económico ha obtenido un porcentaje del 20% ya que el costo es de \$0.00, en la garantía técnica ha obtenido un porcentaje de 5%, en el soporte técnico ha obtenido un 5%, siendo así un porcentaje total para esta alternativa del 100%.



En conclusión se ha determinado que la alternativa más factible, tomando en cuenta parámetros como lo técnico, económico es la alternativa 3 ya que no tiene ningún costo alguno por tal razón es que obtuvo un porcentaje de 100%.

4.7. Descripción De Procesos

LOGIN

Para ingresar al sistema primero se va aparecer una ventana en la cual pida el nombre de usuario y la contraseña, si por alguna razón no introduce bien los datos al momento de poner aceptar se aparecerá un mensaje diciendo que la información no es correcta, esto sucederá tres veces luego de esto el sistema se cerrara.

Ver Anexo N°5: Diagrama De Flujo De Login

INSCRIPCIONES

En este proceso la va ingresa la información del alumno al sistema luego de esto la secretaria genera un documento de inscripción en el cual va estar constando en pago de inscripción si el pago es efectuado se emitirá un recibo de inscripción el cual debe presentar al momento de la matricula

Ver Anexo N°6: Diagrama De Flujo De Inscripción

MATRICULA

En este proceso se procederá a la matriculación del alumno en el cual se pedirá el comprobante de inscripción si tiene el comprobante o recibo de la inscripción se procederá a generar la matricula en el cual se pedirá el pago de la matricula si se efectúa el pago se emitirá un recibo o factura de matrícula caso contrario tendrá que pagar para poder continuar y termina el proceso.

Ver Anexo N°7: Diagrama De Flujo De Matriculación



NOTAS

Aquí el estudiante debe cumplir o rendir pruebas, lecciones, trabajos, exámenes trimestrales, para poder obtener notas caso contrario quedaría sin nota, las notas deben estar en el rango de 1 a 20 si cumple con esto se procederá a hacer una sumatoria de todo para así poder sacar promedio trimestral ya que la institución tiene este método de evaluación, después se procederá a sacar un promedio anual, si este es mayor o igual 13 el alumno aprueba el curso. Si la sumatoria de los tres trimestres es menor a 40 el alumno queda en supletorios. Aquí el alumno debe sacar un puntaje de 12 si la calificación está en el rango de 38 a 39, 14 si está en el rango de 36 a 38, 16 si es de 30 a 35, 18 si es menos de 30 y mayor a 25.

Ver Anexo N°8: Diagrama De Flujo De Notas

4.8. Descripción De Metodología De Desarrollo

En la actualidad, la utilización de metodologías para el desarrollo de aplicaciones es casi imposible omitirla, debido a la gran necesidad de control de variables que conlleva el mismo desarrollo, y para la ordenada elaboración de las aplicaciones, por lo tanto, seguir metodologías y estándares nos llevan a estar en competitividad en todo momento.

Es de suma importancia conocer el modo como se interrelacionan las metodologías con estándares y herramientas siguiendo un único propósito, el cual consiste en la elaboración de aplicaciones de manera eficiente, ordenada y con el menor número de defectos.

No es posible realizar un desarrollo de software de una manera lenta, ya que las exigencias de los clientes actuales conllevan a verse en la necesidad de implementar soluciones rápidas y que cumplan con los requerimientos planteados.

Las metodologías y estándares utilizados en un desarrollo de software nos proporcionan las guías para poder conocer todo el camino a recorrer desde antes



de empezar la implementación, con lo cual se asegura la calidad del producto final, así como también el cumplimiento en la entrega del mismo en un tiempo estipulado.

Es de suma importancia elegir la metodología adecuada, así como las herramientas de implementación adecuadas, es por ello que la metodología RUP basada en UML nos proporciona todas las bases para llevar al éxito la elaboración del software, para ello la utilización de la herramienta Rational Rose es una de las elecciones más acertadas debido a que se fundamenta en el RUP para el desarrollo rápido de aplicaciones.

4.8.1. Metodología de desarrollo aplicada

4.8.1.1. Introducción al RUP

RUP es un proceso de ingeniería de software para modelado orientado a objetos, basado en la notación UML (Unified Modeling Language – Lenguaje Unificado de Modelado). Centrándose en los casos de uso y la arquitectura, es interactivo e incremental. UML es bastante independiente del proceso, lo que significa que se puede utilizar con diferentes procesos de ingeniería del software. El Proceso Unificado de Rational (RUP), es uno de esos enfoques de ciclo de vida que se adapta especialmente bien a UML.

Las características de RUP son:

- Está basado en UML.
- Orientado a sistemas de información de gestión.
- Simple, eficaz y pequeño: fácil de aprender y usar.
- Centrado en el análisis y diseño.
- Permite el desarrollo iterativo e incremental del software.
- Está dirigido por casos de uso y permite la Administración de Requerimientos.
- Uso de Arquitecturas basadas en componentes.
- Modelamiento visual del software.
- Verificación de la calidad del software.



- Control de cambios.
- Ciclos y fases.
- Permite independencia de la Plataforma de Implementación.

El objetivo del Proceso Unificado de Rational es permitir la producción de un software de la mayor calidad que satisfaga las necesidades de los usuarios finales, dentro de planificaciones y presupuestos predecibles. El Proceso Unificado de Rational captura algunas de las mejores prácticas de desarrollo de software, de una forma que es adaptable a un amplio rango de proyectos y organizaciones. En el aspecto de la gestión, el Proceso Unificado de Rational proporciona un enfoque disciplinado sobre cómo asignar tareas y responsabilidades dentro de una organización de desarrollo de software.

4.8.1.2. Dimensiones del RUP

El RUP tiene dos dimensiones:

- El eje horizontal representa tiempo y demuestra los aspectos del ciclo de vida del proceso.
- El eje vertical representa las disciplinas, que agrupan actividades definidas lógicamente por la naturaleza.

La primera dimensión representa el aspecto dinámico del proceso y se expresa en términos de fases, de iteraciones, y la finalización de las fases. La segunda dimensión representa el aspecto estático del proceso: cómo se describe en términos de componentes de proceso, las disciplinas, las actividades, los flujos de trabajo, los artefactos, y los roles.

En la figura se puede observar como varía el énfasis de cada disciplina en un cierto plazo en el tiempo, y durante cada una de las fases. Por ejemplo, en iteraciones tempranas, pasamos más tiempo en requerimientos, y en las últimas iteraciones pasamos más tiempo en poner en práctica la realización del proyecto en sí.

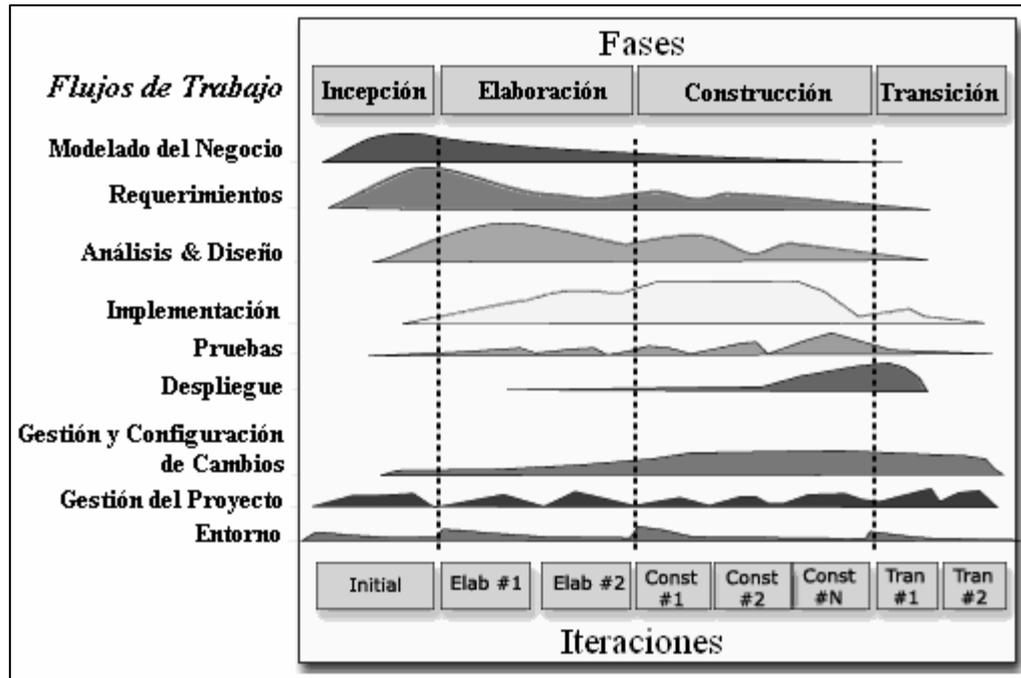


Gráfico N° 16.
 Disciplinas, fases, iteraciones del RUP
 Fuente: Internet

Se puede hacer mención de las tres características esenciales que definen al RUP:

-Proceso Dirigido por los Casos de Uso: Con esto se refiere a la utilización de los Casos de Uso para el desenvolvimiento y desarrollo de las disciplinas con los artefactos, roles y actividades necesarias. Los Casos de Uso son la base para la implementación de las fases y disciplinas del RUP. Un Caso de Uso es una secuencia de pasos a seguir para la realización de un fin o propósito, y se relaciona directamente con los requerimientos, ya que un Caso de Uso es la secuencia de pasos que conlleva la realización e implementación de un Requerimiento planteado por el Cliente.

-Proceso Iterativo e Incremental: Es el modelo utilizado por RUP para el desarrollo de un proyecto de software. Este modelo plantea la implementación del proyecto a realizar en Iteraciones, con lo cual se pueden definir objetivos por cumplir en cada iteración y así poder ir completando todo el proyecto iteración por iteración, con lo cual se tienen varias ventajas, entre ellas se puede mencionar la de tener pequeños avances del proyectos que son entregables al cliente el cual



puede probar mientras se está desarrollando otra iteración del proyecto, con lo cual el proyecto va creciendo hasta completarlo en su totalidad.

-Proceso Centrado en la Arquitectura: Define la Arquitectura de un sistema, y una arquitectura ejecutable construida como un prototipo evolutivo. Arquitectura de un sistema es la organización o estructura de sus partes más relevantes. Una arquitectura ejecutable es una implementación parcial del sistema, construida para demostrar algunas funciones y propiedades. RUP establece refinamientos sucesivos de una arquitectura ejecutable, construida como un prototipo evolutivo.

4.8.1.3. Fases del RUP

Una fase es el intervalo de tiempo entre dos hitos¹ importantes del proceso durante la cual se cumple un conjunto bien definido de objetivos, completan artefactos y se toman las decisiones sobre si pasa a la siguiente fase. El Proceso Unificado de Rational consta de las cuatro fases siguientes:

- **Iniciación** Establece la planificación del proyecto.
- **Elaboración** Establecer un plan para el proyecto y una arquitectura.
- **Construcción** Desarrollar el sistema.
- **Transición** Proporcionar el sistema a sus usuarios finales.

¹ Hito: es una tarea que representa una fecha importante en un proyecto, como la finalización de una fase del proyecto, o la fecha de un informe en particular es vencimiento.

<http://support.microsoft.com/kb/77791/es>

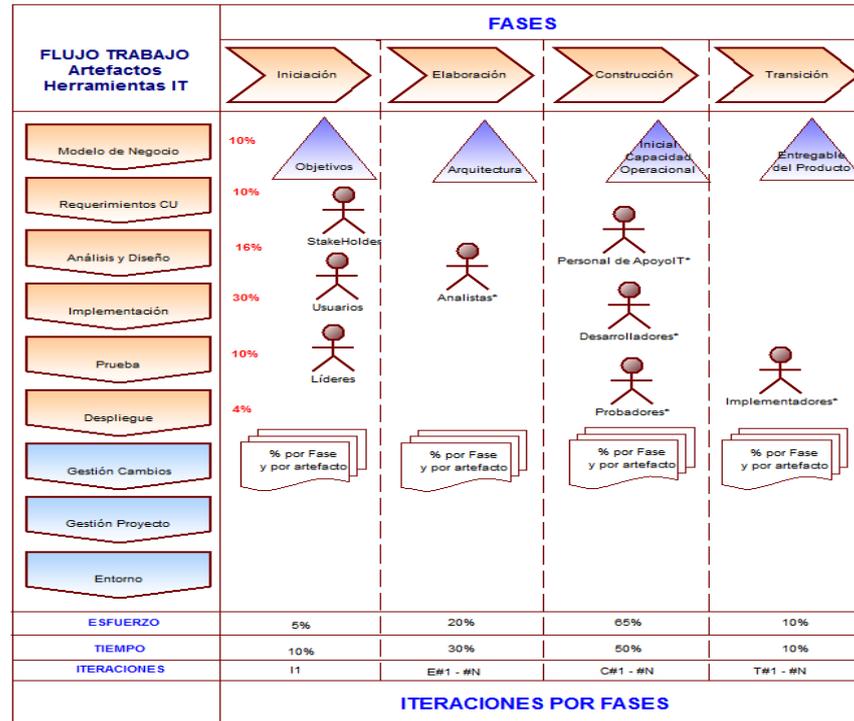


Gráfico Nº 17.
Fases, iteraciones y flujo de trabajo del RUP
Fuente: Internet

Las fases de iniciación y elaboración incluyen las actividades de diseño del ciclo de vida del desarrollo; la construcción y la transición constituyen su producción.

Dentro de cada fase hay varias iteraciones. Una iteración representa un ciclo de desarrollo completo, desde la captura de requisitos en el análisis hasta la implementación y pruebas, que produce como resultado la entrega al cliente o la salida al mercado de un proyecto ejecutable.

Cada fase e iteración se centra en disminuir algún riesgo y concluye con un hito bien definido. La revisión de hitos es el momento adecuado para evaluar cómo se están satisfaciendo los objetivos y si el proyecto necesita ser reestructurado de alguna forma para continuar.

Fase de Inicio: Durante la fase de iniciación, se establece la planificación del proyecto y se delimita su alcance. La planificación del proyecto incluye los criterios de éxito, la evaluación del riesgo, estimaciones de recursos que se necesitarán y un plan de fases que muestre la planificación de los hitos principales. Durante la



iniciación, es frecuentemente crear un prototipo ejecutable que sirva para probar los conceptos.

Al final de la fase de inicio se examinan los objetivos del ciclo de vida del proyecto y se decide si proceder con el desarrollo del sistema.

Fase de Elaboración: Los objetivos de la fase de elaboración son analizar el dominio del problema, establecer una base arquitectónica sólida, desarrollar el plan del proyecto y eliminar los elementos de más alto riesgo del proyecto. Las decisiones arquitectónicas deben tomarse con una comprensión del sistema global. Esto implica que se deben describir la mayoría de los requisitos del sistema. Para verificar la arquitectura, se implementa un sistema que demuestre las distintas posibilidades de la arquitectura y ejecute los casos de uso significativos.

Al final de la fase de elaboración se examinan el alcance y los objetivos del sistema, la elección de la arquitectura y la resolución de los riesgos más grandes, y se decide si se debe pasar a la construcción.

Fase de Construcción: Durante la fase de construcción, se desarrolla de forma iterativa e incremental un producto completo que está preparado para la transición hacia la comunidad de usuarios. Esto implica describir los requisitos restantes y los criterios de aceptación, refinando el diseño y completando la implementación y las pruebas del software.

Al final de la fase de construcción se decide si el software, los lugares donde se instalará y los usuarios están todos preparados para empezar a funcionar.

Fase de Transición: Durante la fase de transición, el software se despliega en la comunidad de usuarios. Una vez que el sistema ha sido puesto en manos de los usuarios finales, a menudo aparecen cuestiones que requieren un desarrollo adicional para ajustar el sistema, corregir algunos problemas no detectados o finalizar algunas características que habían sido pospuestas. Esta fase comienza



normalmente con una versión beta del sistema, que luego podría ser reemplazada con el sistema de producción.

Al final de la fase de transición se decide si se han satisfecho los objetivos del ciclo de vida del proyecto, y se determina si se debería empezar otro ciclo de desarrollo.

4.8.1.4. Disciplinas y Flujos de Trabajo del RUP

Las disciplinas o modelos conllevan los flujos de trabajo, los cuales son una secuencia de pasos para la culminación de cada disciplina, estas disciplinas se dividen en dos grupos: las primarias o de procesos y las de apoyo. Las primarias son las necesarias para la realización de un proyecto de software, aunque para proyectos no muy grandes se pueden omitir algunas; entre ellas se tienen: Modelado del Negocio, Requerimientos, Análisis y Diseño, Implementación, Pruebas, Despliegue. Las de apoyo son las que como su nombre lo indica sirven de apoyo a las primarias y especifican otras características en la realización de un proyecto de software; entre estas se tienen: Entorno, Gestión del Proyecto, Gestión de Configuración y Cambios. A continuación se describe rápidamente cada una de estas disciplinas.

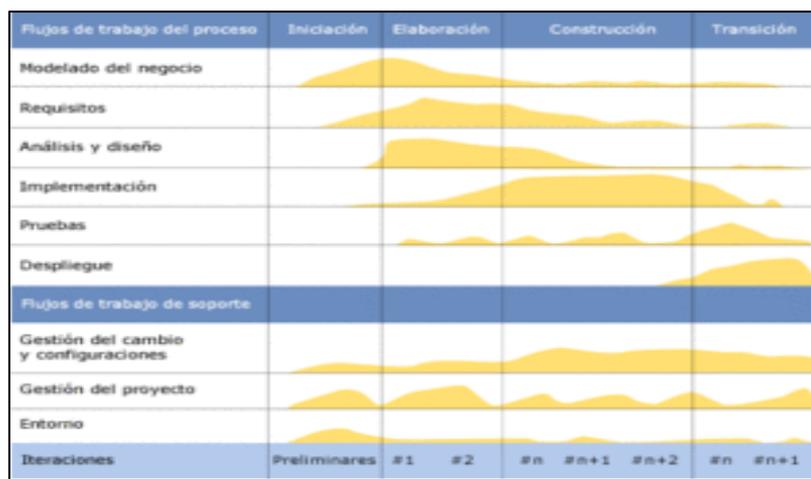


Gráfico N° 18.
Disciplinas y flujo de trabajo del RUP
Fuente: Internet



4.8.1.4.1. Modelado del negocio

Esta disciplina tiene como objetivos comprender la estructura y la dinámica de la organización, comprender problemas actuales e identificar posibles mejoras, comprender los procesos de negocio. Utiliza el Modelo de Casos de Uso del Negocio para describir los procesos del negocio y los clientes, el Modelo de Objetos del Negocio para describir cada Caso de Uso del Negocio con los Trabajadores, además utilizan los Diagramas de Actividad y de Clases.

4.8.1.4.2. Requerimientos

Esta disciplina tiene como objetivos establecer lo que el sistema debe hacer (Especificar Requisitos), definir los límites del sistema, y una interfaz de usuario, realizar una estimación del costo y tiempo de desarrollo. Utiliza el Modelo de Casos de Uso para modelar el Sistema que comprenden los Casos de Uso, Actores y Relaciones, además utiliza los diagramas de Estados de cada Caso de Uso y las especificaciones suplementarias.

4.8.1.4.3. Análisis y diseño

En esta disciplina se definirá la arquitectura del sistema y tiene como objetivos trasladar requisitos en especificaciones de implementación, al decir análisis se refiere a transformar Casos de Uso en clases, y al decir diseño se refiere a refinar el análisis para poder implementar los diagramas de clases de análisis de cada Caso de Uso, los diagramas de colaboración de cada Caso de Uso, el de clases de diseño de cada Caso de Uso, el de secuencia de diseño de Caso de Uso, el de estados de las clases, el modelo de despliegue de la arquitectura.

4.8.1.4.4. Implementación

Esta disciplina tiene como objetivos implementar las clases de diseño como componentes (ej. fichero fuente), asignar los componentes a los nodos, probar los componentes individualmente, integrar los componentes en un sistema ejecutable (enfoque incremental). Utiliza el Modelo de Implementación, conjuntamente los



Diagramas de Componentes para comprender cómo se organizan los Componentes y dependen unos de otros.

4.8.1.4.5. Pruebas

Esta disciplina tiene como objetivos verificar la integración de los componentes (prueba de integración), verificar que todos los requisitos han sido implementados (pruebas del sistema), asegurar que los defectos detectados han sido resueltos antes de la distribución.

4.8.1.4.6. Despliegue

Esta disciplina tiene como objetivos asegurar que el producto está preparado para el cliente, proceder a su entrega y recepción por el cliente. En esta disciplina se realizan las actividades de probar el *software* en su entorno final (Prueba Beta), empaquetarlo, distribuirlo e instalarlo, así como la tarea de enseñar al usuario.

4.8.1.4.7. Gestión y configuración de cambios

Es esencial para controlar el número de artefactos producidos por la cantidad de personal que trabajan en un proyecto conjuntamente. Los controles sobre los cambios son de mucha ayuda ya que evitan confusiones costosas como la compostura de algo que ya se había arreglado etc., y aseguran que los resultados de los artefactos no entren en conflicto con algunos de los siguientes tipos de problemas:

- Actualización simultánea: Es la actualización de algo elaborado con anterioridad, sin saber que alguien más lo está actualizando.
- Notificación limitada: Al realizar alguna modificación, no se deja información sobre lo que se hizo, por lo tanto no se sabe quien, como, y cuando se hizo.
- Versiones múltiples: No saber con exactitud, cual es la última versión, y al final no se tiene un orden sobre que modificaciones se han realizado a las diversas versiones.



4.8.1.4.8. Gestión del proyecto

Su objetivo es equilibrar los objetivos competitivos, administrar el riesgo, y superar restricciones para entregar un producto que satisface las necesidades de ambos clientes con éxito (los que pagan el dinero) y los usuarios. Con la Gestión del Proyecto se logra una mejoría en el manejo de una entrega exitoso de *software*. En resumen su propósito consiste en proveer pautas para:

- Administrar proyectos de *software* intensivos.
- Planear, dirigir personal, ejecutar acciones y supervisar proyectos.
- Administrar el riesgo.

4.8.1.4.9. Entorno

Esta disciplina se enfoca sobre las actividades necesarias para configurar el proceso que engloba el desarrollo de un proyecto y describe las actividades requeridas para el desarrollo de las pautas que apoyan un proyecto. Su propósito es proveer a la organización que desarrollará el *software*, un ambiente en el cual basarse, el cual provee procesos y herramientas para poder desarrollar el *software*.

4.9. Documento De Especificación De Requerimientos

La metodología de desarrollo empleada determina los requerimientos del *software* mediante un documento denominado SRS² (Especificación de requerimientos del *software*)

Ver Anexo N°9: Documento SRS

² SRS Documento que incluye la necesidades del desarrollo de proyectos, sus características y objetivos generales y específicos, actores, stakeholders, roles y responsabilidades.
<http://www.risktechnology.net/metodologia-rup.aspx>



4.10. Modelo Físico

Es el modelo en el cual se examina la vinculación, herencias, dependencias y relaciones de las entidades como por ejemplo en un curso se puede dictar varias materias las cuales heredan el código del curso al que pertenecen.

También llamado integridad referencial es un sistema de reglas que utilizan la mayoría de las bases de datos relacionales para asegurarse que los registros de tablas relacionadas son válidos y que no se borren o cambien datos relacionados de forma accidental produciendo errores de integridad.

Ver Anexo N°10: Modelo Físico

4.11. Diccionario De Datos

Un diccionario de datos es una herramienta de importancia para el administrador de la base de datos, es un catálogo accesible para el usuario de datos relacionados. Con la base de datos.

Con el término de diccionario de datos suele designarse una utilidad de software más general que un catálogo. Los sistemas de diccionario de datos sirven para mantener información relativa al hardware y software, la documentación y los usuarios del sistema, así como otra información pertinente para la administración del sistema

Ver Anexo N°11: Diccionario de datos

4.12. Estándares De Programación

Preámbulo

El desarrollo de software es un proceso exigente y la falta de una metodología estándar y práctica lo ha convertido en un trabajo individual y casi artístico donde cada programador tiene su forma de hacer las cosas.



Principios

La base de esta metodología consiste en fortalecer las actividades de los proyectos de desarrollo de software con procedimientos, documentos y soportes que respalden la correcta ejecución del proyecto.

Un estándar de programación es una forma de “normalizar” la programación de forma tal que al trabajar en un proyecto cualquiera de las personas involucradas en el mismo tenga acceso y comprenda el código. En otras palabras define la escritura y organización del código fuente de un programa. Además de seguir un estándar de programación te facilita como programador la modificación de tu propio código fuente aunque no esté trabajando en equipo.

Por ejemplo para utilizar una determinada nomenclatura para la declaración de variables, o de clases dependiendo de su tipo, te es de gran ayuda porque al leer el nombre de la variable ya sabes con qué tipo de datos estas trabajando y no tienes que buscar la declaración de la variable, todo esto dependiendo siempre del entorno y lenguaje de programación que utilices.

4.12.1. Estándares de Base De Datos

Estándar para las tablas.- para especificar que es una tabla en nuestra base de datos ponemos primero el estándar Tbl y seguido de un guion bajo ponemos el nombre de la tabla con la primera letra en mayúscula.

ESTÁNDAR DE BASE DE DATOS		
OBJETO	ESTÁNDAR	EJEMPLO
Tabla	Tbl	Tbl_Alumnos

Tabla N° 19.

Estándar de base de datos para tablas.

Fuente: Daniel Granizo

A continuación se muestra el estándar de las tablas utilizadas en la base de datos.



ESTÁNDAR DE LAS TABLAS UTILIZADAS	
NOMBRE DE LA TABLA	ESTÁNDAR
Alumno	Tbl_Alumnos
Profesor	Tbl_Profesores
Inscripción	Tbl_Inscripciones
Matriculas	Tbl_Matriculas
Materias	Tbl_Materias
Periodo	Tbl_Periodos
Jornadas	Tbl_Jornadas
Especialidad	Tbl_Especialidades
Nivel	Tbl_Niveles
Transporte	Tbl_Transporte
Ruta del transporte	Tbl_Ruta_Trans
Asistencia	Tbl_Asistencia
Faltas	Tbl_Faltas
Parámetro Periodo	Tbl_ParametrosPeriodo
Tipo Matricula	Tbl_TipoMatricula
Nota	Tbl_Notas
Detalle Notas Año	Tbl_DTNotasAño
Detalle Nota Semestre	Tbl_DTNotasSemestre
Usuario	Tbl_Usuarios
Auditoría	Tbl_Auditoria
Parametrización	Tbl_Parametrizacion

Tabla N° 20.
Estándar de tablas utilizadas
Fuente: Daniel Granizo

Estándar de un atributo de una tabla.- para designar un atributo dentro de la tabla, primero ponemos las tres primeras letras del nombre de la tabla empezando la primera letra con mayúscula y seguido de un guion bajo el nombre del atributo.



ESTÁNDAR DE UN ATRIBUTO DE UNA TABLA		
ATRIBUTO	NOMBRE DE LA TABLA	EJEMPLO
Nombre	Alumnos	Alu_Nombre

Tabla N° 21.
Estándar de un atributo de una tabla.
Fuente: Daniel Granizo.

4.12.2. Estándares de Diseño

Objetos y Controles

ESTÁNDAR DE DISEÑO		
OBJETO	PREFIJO	DESCRIPCIÓN
Label	Lbl	Comentarios en texto
TextBox	Txt	Cajas de texto
List	Lst	Listas
Grid	Grd	Cuadrícula para presentar los datos
ComboBox	Cbo	Selección de datos
Button	Btn	Botón
CheckBox	Chk	Casilla impresa para marcar en ella
RadioButton	Rbtn	Botón de opción
Slider	Sld	Indicador en movimiento
TabControl	Tab	Control de selección de Grd
Border	Brd	Borde
passwordBox	Pass	Control de password

Tabla N° 22.
Estándar de diseño.
Fuente: Daniel Granizo.



4.12.3. Estándares de Programación

ESTÁNDAR DE PROGRAMACIÓN	
VARIABLE	LETRA
Privada	Private
Estática	Static
Publico	Public

Tabla N° 23.
Ejemplo de variables.
Fuente: Daniel Granizo.

Tipos de datos utilizados

ESTÁNDAR DE TIPO DE DATOS UTILIZADOS	
TIPO DE DATO	ESTÁNDAR
Binario	Bit
Carácter	Char
Fecha Tiempo	DateTime
Numero Decimal	Decimal
Flotante	Float
Figura	Image
Entero	Integer
Numérico	Numeric
Fecha Corta	SmallDateTime
Entero Corto	Smallint
Varios caracteres	VarChar

Tabla N° 24.
Estándar de tipo de datos utilizados.
Fuente: Daniel Granizo.



4.13. Documento De Diseño

Mediante el documento SDS³, se detallara el diseño de la interfaz, este a su vez surge de los diagramas de casos de uso elaborados en la investigación.

Ver Anexo N°12: Documento de diseño (SDS)

4.14. Pruebas Y Depuración

Las pruebas están diseñadas para depurar la mayor cantidad de errores en la aplicación por que se considerarán las clases y métodos más representativos del negocio.

Para la realización de las pruebas se utilizó la técnica de la “Caja Negra”⁴. Se analizan las entradas y las salidas de los métodos, sin analizar qué ocurre dentro de los mismos.

4.14.1. Pruebas de Integración

PRUEBAS DE INTEGRACIÓN	
Objetivo de la Prueba	Asegurar que las llamadas a métodos entre componentes, se realicen en forma apropiada guardando la integridad de información con la base de datos.
Técnica:	Invocar a cada método o proceso mediante consultas o ingreso de datos válidos e inválidos hacia la aplicación. Inspeccionar la base de datos para asegurar que: <ul style="list-style-type: none">• Los datos han sido introducidos correctamente.• Todas las funciones de la base de datos se ejecutan apropiadamente.• Revisar que el retorno o recuperación de datos se efectúan adecuadamente.

³ SDS establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis hacia una de diseño. <http://www.risktechnology.net/metodologia-rup.aspx>

⁴ Técnica de la Caja Negra se llevan a cabo sobre la interfaz del software, obviando el comportamiento interno y la estructura del programa. <http://indalog.ual.es/mtorres/LP/Prueba.pdf>



Criterios de Terminación	Todos los métodos y procesos de acceso a la base de datos funcionan de acuerdo al diseño y sin ninguna inconsistencia de datos.
Consideraciones especiales:	<p>El gestor de base de datos utilizado para el ingreso o modificación de datos es SQL Server 2008 Express, bajo plataforma Windows 7.</p> <p>Se necesitará de una conexión a la base de datos y un analizador de consultas para probar la base de datos.</p> <p>El ingreso del ID del dispositivo y fechas relacionadas con el ingreso de dispositivos se generan automáticamente.</p>

Tabla N° 25.

Prueba de integración.

Fuente: Daniel Granizo.

4.14.2. Pruebas de Validación

PRUEBAS DE VALIDACIÓN	
Objetivo de la Prueba	Asegurar que todas las cajas de texto estén validadas, ya sean para poder introducir solo texto, números, o los dos a la vez.
Técnica:	Ingresar datos en las cajas de texto con el fin de comprobar si están validadas, si la información a introducir es solo texto y el usuario ingresa números, saldrá un mensaje el cual diga solo se permiten texto de igual manera sucederá con la información que es solo para números, sino que en el mensaje saldrá solo se permite números.
Criterios de Terminación	Todas las cajas de texto donde que utilice el sistema deben estar validados caso contrario hay que validarlas.
Consideraciones especiales:	Al momento de ingresar la dirección de correo electrónico hay que tener en cuenta que el correo si exista y solo me permita ingresar direcciones electrónicas.

Tabla N° 26.

Prueba de validación.

Fuente: Daniel Granizo.



4.14.3. Pruebas de Seguridad

PRUEBAS DE SEGURIDAD	
Objetivo de la Prueba	Comprobar la seguridad del sistema
Técnica:	<p>Para la realización de esta prueba se hará lo siguiente:</p> <ul style="list-style-type: none"> • Se intentara ingresar al sistema con un usuario y password no registrados en el sistema. • Se intentara inyectar SQL desde cualquier caja de texto.
Criterio de Terminación:	<p>La prueba termina con éxito cuando:</p> <ul style="list-style-type: none"> • No se logre ingresar a las opciones del sistema sin que el usuario está registrado. • Cuando no se logre inyectar SQL.
Consideraciones especiales:	

Tabla N° 27.

Pruebas de Seguridad

Fuente: Daniel Granizo.

EJEMPLO DE PRUEBA DE SEGURIDAD			
Nº	Acción	Resultado	Éxito
1	Ingresar al sistema con un nombre de usuario y clave otorgado	Ingreso normal al sistema	SI
2	Salir del sistema	El sistema vacía toda la información de la sesión	SI
3	Tratar de inyectar SQL	El sistema valida el tamaño de las cajas de texto y no permite la inyección de SQL	SI

Tabla N° 28.

Ejemplo de Pruebas de Seguridad

Fuente: Daniel Granizo.



4.14.4. Pruebas de interfaces graficas de usuario

PRUBAS DE INTERFACES GRAFICAS	
Objetivo de la Prueba	Comprobar que la interfaz gráfica sea amigable para el usuario y fácil de manejar.
Técnica:	Verificar que la interfaz no se distorsione al momento de maximizarla o minimizarla. Se le pedirá al usuario que compruebe si la interfaz gráfica es amigable, fácil de manejar y entendible.
Criterio de Terminación:	Que la interfaz aparezca distorsionada.
Consideraciones especiales:	Ninguna.

Tabla Nº 29.

Pruebas de interfaz gráfica.

Fuente: Daniel Granizo.

4.15. Instalación Del Sistema

Incorporar en una computadora una aplicación o un dispositivo para poder ser utilizado.

Las aplicaciones más elaboradas suelen tener un programa instalador que facilita el proceso.

El instalador del sistema será un setup en donde nosotros al ejecutarlo permitirá instalar todos los componentes a utilizar de la aplicación.

4.16. Recopilación Y Carga De Datos

Al momento que el sistema esté instalado y funcionando deberemos instalar el Sql Server 2008 para el ingreso de los datos y tener la seguridad que la información va a estar en un lugar seguro.

La carga de datos se lo realizara en la aplicación SISCONACA pero el que podrá ingresarlos, será exclusivamente el administrador del sistema, con su perfil el cual puede acceder a la opción de administrador del sistema para llenar los campos requeridos por los usuarios.



4.17. Pruebas Y Depuración Final En Funcionamiento

Se realizó y se probó la base de datos diseñada para el colegio, se hizo pruebas con ingresos de datos de alumnos, profesores, inscripciones, matriculas, mantenimientos a los usuarios, alumnos, profesores, para verificar que el sistema esté en óptimas condiciones.

Además se empezó con la instrucción al personal que sería el responsable del manejo del sistema para que empiece a familiarizarse con el mismo, a fin de evitar inconvenientes a futuro. Una vez purificado los posibles erros, se procedió a almacenar la información real en la base de datos para empezar con el control total de los inventarios y sus diferentes movimientos, dotando así al colegio de una herramienta muy útil.

4.18. Puesta En Marcha Del Sistema

La visualización en vivo para el usuario de (SISCONACA) entrará en vigencias desde el mes de Junio del 2012, con las configuraciones aprobadas por el rector del colegio.

Una vez instalado el programa se procederá a la carga de los datos reales de dicha institución, mediante el ingreso de la información parametrizada para el óptimo funcionamiento del sistema.

4.19. Capacitación Al Usuario Final

Se realizó a los usuarios seleccionados, una capacitación objetiva de las principales funcionalidades del sistema; se explicó su funcionamiento, así como los formularios que se presentan en la aplicación, los mismos que corresponden a los formularios utilizados manualmente para llevar el proceso de inscripciones, matrículas y reporte de nota.

En la capacitación se tomó en cuenta las partes fundamentales del sistema, como son: El realizar ingresos de información, Realizar el proceso de notas, y la emisión de consultas y reportes; también se explicó el tipo de menús utilizado y los perfiles que permiten su acceso.



En la explicación de funcionamiento se realizó ejemplos con cada una de las interfaces que presenta el sistema, para los perfiles que maneja el sistema, también se analizó las respectivas relaciones entre los módulos existentes.

Los ejemplos se realizaron desde el inicio del sistema, ingreso de parámetros iniciales, ingreso de datos requeridos para el proceso de inscripciones, matrículas y de notas como son:

El ingreso de alumnos, profesores, la especialidad, jornada, curso o nivel, y las respectivas calificaciones del alumno para el proceso de notas, terminando con la impresión del reporte de notas.

En la evaluación del sistema, se va a tomar los siguientes aspectos:

Se utilizaron las técnicas de Conferencia, y Audiovisuales, para poder abarcar a toda la población involucrada en el uso del nuevo sistema, además de la técnica de Instrucción Directa Sobre el puesto, misma que se imparte durante las horas de trabajo. Se emplea básicamente para asignar a obreros y empleados a desempeñar un puesto actual. La instrucción es impartida por un capacitador, supervisor o compañero de trabajo. En la mayoría de los casos el interés del capacitador se centra en obtener un determinado producto y en una buena técnica de capacitación.

Se distinguen varias etapas:

Se brinda a la persona que va a recibir la capacitación, una descripción general del puesto, su objetivo, y los resultados que se esperan de él.

El capacitador efectúa el trabajo a fin de proporcionar un modelo que se pueda copiar.

Se pide al individuo que imite el ejemplo. Las demostraciones y las prácticas se repiten hasta que la persona domine la técnica.

Se pide a la persona que lleve a cabo el ejercicio sin supervisión.



4.20. Capacitación Al Personal Técnico

Capacitación del Sistema al Personal Técnico

4.20.1. Datos Informáticos

Empresa:	Colegio Dr. M. B. C. M.
Número de Trabajadores de la Empresa:	1
Nombre Capacitador:	Daniel Granizo.
Material Entregable:	Manual Técnico del Sistema.
Método de Capacitación:	Charla.
Tiempo de Capacitación:	8 horas.

4.20.2. Desempeños Esperados

LO QUE EL PERSONAL TÉCNICO ALCANZARÁ HACER

Al término de la capacitación del Sistema de Control Académico, el técnico (s) del colegio, estará en la capacidad de manejar eficientemente los siguientes puntos:

- Entender el funcionamiento del Sistema.
- Ejecutar el sistema con su respectiva clave de administrador.
- Lograr Ingresar al mantenimiento de todas las entidades.
- Realizar la respectiva auditoria del Sistema.
- Monitorear la Base de Datos.

LO QUE EL PERSONAL TÉCNICO DEBE SABER HACER

Para poder iniciar la ejecución del Sistema, el personal técnico previamente debe conocer y saber usar los siguientes tópicos:

- Manejo de Base de Datos
- Programación en Visual Studio 2010.
- SQL Server 2008.
- Manual de Procesos y Procedimientos



- Auditoria de Sistemas.

4.20.3. Diseño Operativo

ACTIVIDAD	RECURSOS	RESULTADOS
Charla Audiovisual: Revisión del funcionamiento del Sistema, Base de Datos, Metodología de Desarrollo, Plataforma de Desarrollo.	Manual Técnico, Diapositivas: Mapas Conceptuales,	Participar con argumentos relevantes. Resolver inquietudes de otros compañeros.
Laboratorio: Demostración del Sistema	Laboratorio Computadoras Software	Vincular al personal técnico con el Sistema.

Tabla N° 30

Actividades Capacitación a Personal Técnico

Fuente: Daniel Granizo

4.20.4. Observaciones y Recomendaciones

- El personal técnico deberá asistir a la charla revisando el Manual Técnico antes entregado por parte del Colegio.
- Participación analítica y crítica en la charla.

Ver Anexo N°13: Manual Técnico



CAPITULO V

ANÁLISIS DE IMPACTOS

5.1. Principales Impactos

5.1.1. Científico.

La investigación desarrollada por el proyecto tiene una característica especial ya que aplicativos y tecnologías implementadas en nuestro medio únicamente se lo realizaba con software extranjero de los cuales no se podía tener el soporte técnico necesario, por lo tanto al tratarse de una implementación local se podrá beneficiar los técnicos y especialistas, que se dedican a este tipo de negocios ya que cualquier inquietud y soporte será localmente.

5.1.2. Educativo.

El desarrollo de un sistema informático involucra el conocimiento del diseño y la programación, por lo tanto al realizar un proyecto de fusión de tecnologías se dará un aporte muy importante para el conocimiento de los alumnos de la institución que se encuentran en los niveles inferiores los cuales se pueden motivar y desarrollar proyectos de fusión de muchas tecnologías existentes.

5.1.3. Técnico.

Para el desarrollo del sistema se tomó en cuenta varias herramientas las cuales nos sirvieron para ir desarrollando paso a paso nuestro sistema, dentro de estas, esta Rational Rose que nos ayudó hacer los diferentes diagramas mencionados en el capítulo anterior, Microsoft Expression Blend nos ayudó en el diseño de la interfaz para el usuario, Visual Studio 2010 IDE de desarrollo el cual soporta varios



lenguajes de programación como lo c# el cual se utilizó para el desarrollo en tres capas.

La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

Lo referente a este tema se considera las herramientas que se utilizaron para el modelamiento y desarrollo del Sistema de Control Académico, considerando que las mismas fueron estudiadas en las aulas de la Institución, en consecuencia su aplicación no tuvo mayor dificultad.

5.1.4. Tecnológico.

El ejecutar temas de innovación tecnológica aplicando los conocimientos recibidos en las aulas hace que cualquier proyecto sea considerado como un aporte institucional hacia la sociedad, mejor aún si lo que se resta realizando es la unión de varias tecnologías en un solo aplicativo, que es producto de la investigación tecnológica.

5.1.5. Empresarial.

Siendo la visión emprendedora la realización en un futuro de una empresa, se ha cristalizado el primer paso que consiste en tener un aplicativo que maneje las reglas del negocio, por lo tanto a medida que se perfecciona el mismo se continuara con una mentalidad empresarial.

5.1.6. Social.

La implementación de este sistema y la estructuración de una empresa en un futuro, para brindar este tipo de servicio, lograra crear fuentes de trabajo tanto en



el ámbito administrativo como técnico, por consiguiente se estará brindando un aporte al ámbito social.

5.1.7. Económico.

La generación de impuestos por el servicio brindado con este tipo de empresa beneficiara a la economía nacional ya que su aporte será con todos lo que establece la ley en pagos de aranceles e impuestos en general.

5.2. Conclusiones

1. La metodología de investigación científica aplicada al proyecto me permitió determinar la utilización de tipos de investigación, métodos y herramientas que facilitaron la obtención de resultados cuantitativos y cualitativos, que facilitaron la obtención de procedimientos, tareas y procesos propios de la toma de decisiones del Sistema de Control Académico.
2. La aplicabilidad de los pasos que se siguen en ingeniería de software fueron determinantes ya que mediante el análisis me permitieron discernir todo el flujo de información que se ejecuta en el proceso de control académico de interés, al mismo tiempo poder diseñar adecuadamente los procesos sus relaciones, las bases de datos y sus objetos, a fin de poder obtener una organización metódica y bien estructurada del Sistema de Control Académico.
3. Los conocimientos adquiridos en el Instituto tecnológico Superior Cordillera, me permitieron consolidar las teorías pragmáticas en herramientas de ejecución prácticas utilizando lenguajes de programación, y bases de datos materializadas en conocimientos de programación informática. Lo que definió en una concatenación de ideas lógicamente estructuradas y enlazadas hacia la consecución de un objetivo general “Analizar, desarrollar e implementar un Sistema de Control Académico.
4. En un entorno de desarrollo de software se debe realizar muchas pruebas de campo como validaciones de sistema, validaciones de usuarios, para poder decir que el software está terminado y en funcionamiento, más aun si



consideramos que en nuestro caso también se tiene que mantener una especial atención, lo que permitirá la consolidación de la información.

5.3. Recomendaciones

1. Todo trabajo de investigación debe sustentarse con una metodología métodos y herramientas de la investigación científica por lo tanto es recomendable que las personas se preparen en este campo para realizar cualquier investigación, más aun considerando que esto es la base para poder realizar el proyecto de grado previo a la obtención del título de Tecnólogo Analista de Sistemas.
2. Seguir la misma secuencia de desarrollo de software o quizá una mejor a fin de poder llegar a obtener un sistema completo, aplicable y amigable, es el resultado de haber seguido una metodología de desarrollo y técnicas de programación que permita interactuar con los datos alfanuméricos (información), a fin de poder estructurar un sistema real de Toma de decisiones.



CAPITULO VI

6.1. Cronograma de Actividades

Ver Anexo N°14: Cronograma de actividades.

6.2. Glosario de Términos

Abstracción

1. Identificación de las características esenciales de una cosa que la distinguen de otras cosas. Una abstracción siempre implica el punto de vista o perspectiva de quien la realiza.
2. Tipo de dependencia que relaciona dos elementos que representa el mismo concepto en diferentes niveles de abstracción.

Acción

Una acción es una computación atómica ejecutable que produce un cambio en el estado del sistema, o el retorno de un valor. Una acción es una actividad que no se puede descomponer más. Representa un paso dentro de un algoritmo.

Actividad

Es una ejecución no atómica en curso, dentro de una máquina de estados. Una actividad produce alguna acción, compuesta de computaciones atómicas ejecutables que producen un cambio en el estado del sistema, o el retorno de un valor.



Una actividad representa la realización de una o varias tareas que causan un cambio en el estado del sistema. Puede estar compuesta por otras actividades y por acciones, y se puede descomponer en otros diagramas de actividad.

Actor

Es una entidad externa que interactúa con el sistema que se está modelando.

Agregación

Se presenta cuando se modela una relación <<todo/parte>>, en la que una clase representa algo grande (el <<todo>>) que está formado por elementos más pequeños (las <<partes>>).

Arquitectura de software.- Una Arquitectura de Software, también denominada *Arquitectura lógica*, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información.

Artefacto

Un artefacto es una pieza de información que es utilizada o producida mediante un proceso de desarrollo de software.

Asociación

Es una relación estructural que describe un conjunto de enlaces o conexiones entre objetos.

Atributo

Es la descripción de una característica estructural de una clase en su nivel abstracto.



Base De Datos

Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.

Caso de uso

Es una descripción de una secuencia de acciones que un sistema ejecuta para producir un resultado observable de interés para un actor. Un caso de uso se utiliza para definir la estructura de los aspectos de comportamiento en un modelo.

Clase

Es un conjunto de datos (variables o campos) y de las funciones (los métodos) utilizadas para acceder a estos datos. A los datos y funciones pertenecientes a una clase se les conoce como variables y funciones o método miembro.

Es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones y semántica.

Clase abstracta

Dentro de una jerarquía de clase es frecuente especificar que algunas clases son abstractas; es decir, que no pueden tener instancias directas. Una clase abstracta solo puede tener instancias a través de sus clases derivadas o subclasses.

Clase asociación

La clase asociación se presenta cuando la propia relación de asociación entre dos clases tiene propiedades. Por ejemplo, en una relación empleado/ empleador entre una empresa y una persona, hay una clase trabajo que representa las propiedades de esta relación y que se aplica exactamente a un par de Empresa y Persona.



Colaboración

Es una sociedad de elementos y roles que colaboran entre sí para conseguir un comportamiento mayor a la suma de los comportamientos de los elementos. Define una interacción entre los elementos.

La colaboración es un elemento que tiene tanto dimensión estructural como de comportamiento.

Componente

Es una parte física y reemplazable de un sistema que se corresponde con un conjunto de interfaces y lo implementa. Es típicamente el empaquetamiento físico de distintos elementos lógicos como clases, interfaces y colaboraciones.

Constructor

Es un método de una clase que se le utiliza para crear una instancia u ocurrencia de la clase.

Dependencia

Es una relación semántica entre dos elementos en la que un cambio en uno de ellos (el elemento independiente) puede afectar a la semántica del otro elemento (el elemento dependiente).

Desarrollo iterativo

Es un esquema de desarrollo de software en el que se pasa por las etapas de análisis, diseño, codificación, prueba y despliegue de manera iterativa a través diferentes etapas y en diferentes instantes del proyecto.

Diagrama de actividades

Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema. Es una vista simplificada de lo que ocurre en un sistema.



Diagrama de casos de uso

Representa los requerimientos de los usuarios del sistema y especifica la funcionalidad y comportamiento de los elementos del sistema cuando interactúan entre ellos. Estos diagramas son importantes en el modelado y organización del comportamiento de un sistema.

Diagrama de clase

Muestra un conjunto de clases, interfaces, colaboraciones y sus relaciones.

Representan la vista de diseño estática de un sistema, y son los diagramas más comunes en el modelado de sistemas orientados u objetos.

Diagrama de colaboración

Es un diagrama de interacción que muestra La organización estructural de los objetos que envían y reciben mensajes.

Diagrama de componentes

Muestra la organización y la dependencia entre los componentes de un sistema. Cubre la vista de implementación estática de un sistema.

Diagrama de despliegue

Muestra la topología hardware sobre la que se ejecuta el sistema y los componentes que residen en cada uno de los elementos del hardware.

Diagrama de estados

Muestra los estados por los que pasa un objeto, y los eventos que provocan las transiciones para que el objeto pase de estado a otro.



Diagrama de objetos

Muestra un conjunto de objetos y sus relaciones, al igual que el diagrama de clases representa la vista de diseño estática de un sistema, pero desde el punto de vista de casos reales o prototipos mostrando instancias de las clases representadas en el diagrama de clases.

Diagrama de secuencia

Es un diagrama de interacción que cubre la vista dinámica de un sistema. Un diagrama de interacción muestra un conjunto de objetos, sus relaciones, y los mensajes intercambiados entre ellos. El diagrama de secuencia muestra el orden cronológico de los mensajes intercambiados entre los objetos.

Encapsulación

Este término se refiere al hecho de que los datos (variable), y el código que manipula dichos datos (métodos), se encuentran <<encerrados>> dentro del objeto.

Enlace

Es una conexión semántica entre dos objetos, es decir es un a instancia de una asociación entre dos objetos. Cuando existe un enlace, un objeto puede enviarle un mensaje al otro.

Escalable

Se dice que un sistema es escalable cuando su diseño e implementación se ha establecido de modo tal que puede crecer o escalar a un sistema mayor ya sea en hardware o en software.

Escenario

Se conoce como escenario a una instancia de un caso de uso. Un escenario es utilizado para describir como los casos de uso interactúan entre grupos de objetos y permite



identificar los objetos las clases y las interacciones entre objetos que son necesarios para conseguir la funcionalidad especificada.

Estado

Es una condición o situación en la vida de un objeto durante la cual satisface alguna condición, realiza alguna actividad o espera algún evento. Un estado no es instantáneo sino que dura una cantidad de tiempo finita.

Estereotipo

Un estereotipo representa un a variación de algún elemento existente dentro del modelo, con la misma forma (tal como los atributos y las relaciones) pero con una intención diferente.

Evento

Es la especificación de un acontecimiento significativo que ocupa un lugar en el tiempo y en el espacio. En el contexto de las máquinas de estados, un evento es la aparición de un estímulo que puede disparar una transición de estado.

Flujo de control

En un sistema de control hay un único flujo de control, lo que significa que en un momento dado solo puede tener lugar a una sola cosa.

Flujo de eventos

El flujo de eventos de un caso de uso describe lo que el sistema debe hacer, y no como lo debe hacer.

El flujo de eventos es una lista de los pasos que sigue un actor durante su interacción con el sistema.



Foco de control

En un diagrama de secuencia, el rectángulo delgado y en posición vertical debajo representa el foco de control y define el periodo durante el cual el objeto ejecuta una acción, ya sea directamente o a través de un procedimiento subordinado. La parte superior se alinea con el inicio de la acción y la parte inferior con el fin de la acción pudiendo incluir un mensaje de retorno.

Flujo de objetos

Un objeto puede cambiar de estado por su participación en una actividad, y pasar a otra actividad para participar en otro conjunto de acciones que puede o no cambiar su estado. Por tanto existe un flujo de objetos, pues este se traslada por diferentes actividades cambiando su estado.

Generalización

Es una relación de especialización/generalización en la que los objetos del elemento especializado (el hijo) pueden sustituir a los objetos del elemento general (el padre). El hijo hereda la estructura y el comportamiento del padre.

Herencia

Se entiende por herencia a la capacidad de un objeto derivado (objeto hijo) de heredar las características del objeto base (objeto padre). El objeto derivado viene a ser una especialización del objeto base, pudiendo redefinirse determinados métodos y añadirse nuevos métodos y propiedades que solo estarán disponibles en el objeto derivado.

Instancia

Una instancia es una ocurrencia de un clasificador tal como una clase, un caso de uso, un proceso, etc.



Interacción

Es un comportamiento que comprende un conjunto de mensajes intercambiados entre un conjunto de objetos dentro de un contexto particular.

Interface, Interfaz

Es una colección de operaciones que significa el comportamiento de una clase o componente. Describe el comportamiento de la clase o componente que es visible externamente.

Línea de vida

En un diagrama de secuencia, es una línea vertical discontinua bajo el objeto que representa la existencia del objeto a lo largo del tiempo de ejecución de la interacción.

Máquina de estados

Es un comportamiento que especifica la secuencia de estados por los que pasa un objeto o una interacción durante su vida, en respuesta a eventos, junto con las reacciones o estos eventos.

Mensaje

Los objetos que se encuentran conectados por enlaces pueden intercambiar mensajes, enviar eventos e invocar operaciones. Un mensaje es la especificación de una comunicación entre objetos con la expectativa de que se desencadene una actividad como consecuencia del mensaje. La recepción del mensaje por el otro objeto genera una instancia de un evento en él, que puede resultar en una acción cuya resultante es la ejecución de un procedimiento computacional esta acción puede producir un cambio de estado del objeto.

Método

Un método es un procedimiento que implementa una operación de una clase.



Modelado

Arte de construcción de modelos. Véase Modelo

Modelo

Un modelo es una representación de la realidad en la que se toma en cuenta solo los detalles relevantes con la finalidad de obtener la solución de un problema.

Modelo del dominio

Un modelo del dominio captura los conceptos (tipo de objetos o clases) más importantes en el contexto del sistema. Los objetos del dominio representan las cosas que existen o los eventos que ocurren en el entorno en el que trabaja el sistema.

Multiplicidad

Indica la cantidad de instancias u ocurrencias que un elemento puede tener.

Nodo

Es un elemento físico que existe en un tiempo de ejecución y representa un recurso computacional, que por lo general dispone de memoria y en ocasiones de capacidad de procesamiento.

Objeto

Un objeto es una instancia u ocurrencia concreta de la abstracción que representa la clase.

Operación

Es la especificación de una transformación o consulta que puede ejecutar un objeto. Un método es un procesamiento que implementa una operación.



Paquete

Es un mecanismo de propósito general para organizar los elementos en grupos. Un paquete es un elemento conceptual, solo existe en tiempo de desarrollo.

Polimorfismo

Este término se refiere al hecho de que algo puede tener muchas formas. En la programación orientada a objetos se refiere a la posibilidad de que un método puede tener un nombre ABC, pero adoptar múltiples formas (distintos comportamientos). El polimorfismo permite redefinir método en los objetos derivados que sobrescriban a los métodos del objeto clase.

Procesador

Es cualquier dispositivo físico con capacidad de procesamiento y que adicionalmente puede tener capacidad de almacenamiento.

Proceso

Es un flujo que puede ejecutarse concurrentemente con otros procesos.

Relación de asociación

Esta relación se presenta entre un actor y un caso de uso y representa la comunicación entre el actor y el caso de uso.

Programación por capas

Es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.



Relación de dependencia

Una dependencia es una relación de uso que declara que un cambio en la especificación de una clase puede afectar a otra clase que la utiliza.

Relación de extensión

Esta relación se presenta cuando la funcionalidad definida en un caso de uso es invocada por otro caso de uso, pero solo bajo ciertas condiciones. Se emplea para modelar un caso de uso en el que una parte de la funcionalidad es opcional, de este modo se puede separar la parte de la funcionalidad que siempre se llevara a cabo de la parte del a funcionalidad que es opcional.

Relación de inclusión

Esta relación se presenta cuando varios casos de uso tienen un aparte cuya funcionalidad es común.

Responsabilidad

Las clases deben cumplir cierta labor. Cuando hablamos de orientación a objetos, a dicha labor se le conoce como responsabilidades de la clase. Una responsabilidad es un contrato u obligación de la clase.

Rol

Representa el papel específico que juega una clase en la relación de asociación. Es la cara que la clase de un extremo de la relación presenta a la clase en el otro extremo.

Secuencia

Es un flujo de mensajes que se inicia en un objeto y puede involucrar a varios objetos. El inicio de una secuencia tiene su origen en algún proceso o hilo y la secuencia continuara mientras el proceso o hilo que la contiene exista.



Subsistema

Es un paquete de elementos que se trata como una unidad incluyendo una especificación de contenido del paquete. Un subsistema tiene un conjunto de interfaces que describen su relación con el resto del sistema y las circunstancias en las que se puede utilizar.

Tipo de dato

Es un descriptor de un conjunto de valores que carece de identidad (no tiene existencia independiente ni posibilidad de efectos secundarios). Incluye a los tipos primitivos predefinidos como números y cadenas, así como los tipos enumerados como los booleanos.

Topología

Define la arquitectura física o de hardware de un sistema, es decir, la forma como están conectados los diferentes elementos que conforman el hardware.

Transición

Cuando se completa la ejecución de una actividad o una acción, el flujo de control pasa inmediatamente al siguiente estado de acción o de actividad. El flujo se especifica con transiciones que muestran el cambio desde un estado de actividad o acción al siguiente.

Transición interna

Es aquella que posee una acción pero no implica un cambio de estado.

Vista de casos de uso

Esta vista proporciona un enfoque de alto nivel que nos permite entender lo que el sistema hará sin preocuparnos de los detalles de cómo lo hará. La vista de casos de uso incluye a todos los casos de uso y los diagramas de casos de uso que forman parte del sistema.



En Rational Rose describe el comportamiento del sistema desde el punto de vista de los usuarios finales, analistas y personal encargado de las pruebas del sistema.

Vista de componentes

En Rational Rose la vista de componentes contiene información sobre los componentes, entendidos estos como módulos físicos de código: bibliotecas de código, archivos ejecutables, bibliotecas runtime, etc.

Vista de despliegue

Muestra los nodos que forma la topología hardware sobre la que se ejecuta el sistema. Esta vista se centra en la distribución, entrega e instalación de las partes que forman el sistema físico.

En Rational Rose la vista de despliegue muestra el despliegue físico del sistema. Esta vista se concentra en la representación de los procesos y del hardware del sistema.

Vista de diseño

Muestra los requisitos funcionales del sistema, es decir, los servicios que el sistema brinda a los usuarios finales. Comprende a las clases, interfaces y colaboraciones que forman la solución del sistema.

Vista de implementación

Muestra los componentes y archivos necesarios para hacer disponible el sistema físico. Esta vista se centra principalmente en la gestión de la configuración de las distintas versiones del sistema.

Vista de procesos

Muestra los hilos y procesos que forman los mecanismos de sincronización y concurrencia del sistema. Comprende la funcionalidad, capacidad de crecimiento y rendimiento del



sistema. Los diagramas utilizados ponen énfasis en las clases activas que representan a los hilos y procesos.

Vista lógica

En Rational Rose la vista lógica nos muestra una representación detallada de las partes del sistema y cómo interactúan entre ellas. Esta vista se enfoca en cómo se implementara el comportamiento del sistema en cada uno de los casos de uso.

6.3. Bibliografía

Federico Lemus, (1989), Investigación Científica, Edit: Málaga, Edic: 14, Madrid España.

Marco Caldas, (1999), Preparación y evaluación de proyectos, Edit: Amazonas, Edic: 01, Quito Ecuador.

Ing. Sergio Matsukawa Maeda, (2004), Analisis y Diseño Orientado a Objetos con UML y Rational Rose, Edit: Macro, Edic: 01, Lima-Peru.

6.4. Netgrafía

Topología Estrella

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

Rational Rose

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

Definición De Blend

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

Visio

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

Programación Por Capas

http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas



Curso WPF para programadores

http://www.clikear.com/Curso_WPF_programadores_16297.aspx



ANEXOS



Anexo N°1

Formato de Encuesta



ENCUESTA

RESPONDA LAS SIGUIENTES PREGUNTAS PONIENDO UNA X EN LOS CASILLEROS DE RESPUESTA DE CADA PREGUNTA

1. ¿Sabe si el colegio posee controles automatizados para el manejo de inscripciones, matrículas y notas?

SI

NO

2. ¿Cree usted que la atención a los padres de familia y alumnado mejoraría con la implantación de un sistema de control académico?

SI

NO

3. ¿Conoce usted el sistema de evaluación que se utiliza en el colegio?

SI

NO

4. ¿Considera usted que el sistema de evaluación del colegio es bueno?

SI

NO

5. ¿Al momento de las inscripciones y matrículas la atención es buena?

SI

NO

6. ¿Con la creación del sistema se facilitara el trabajo de la secretaria?

SI

NO

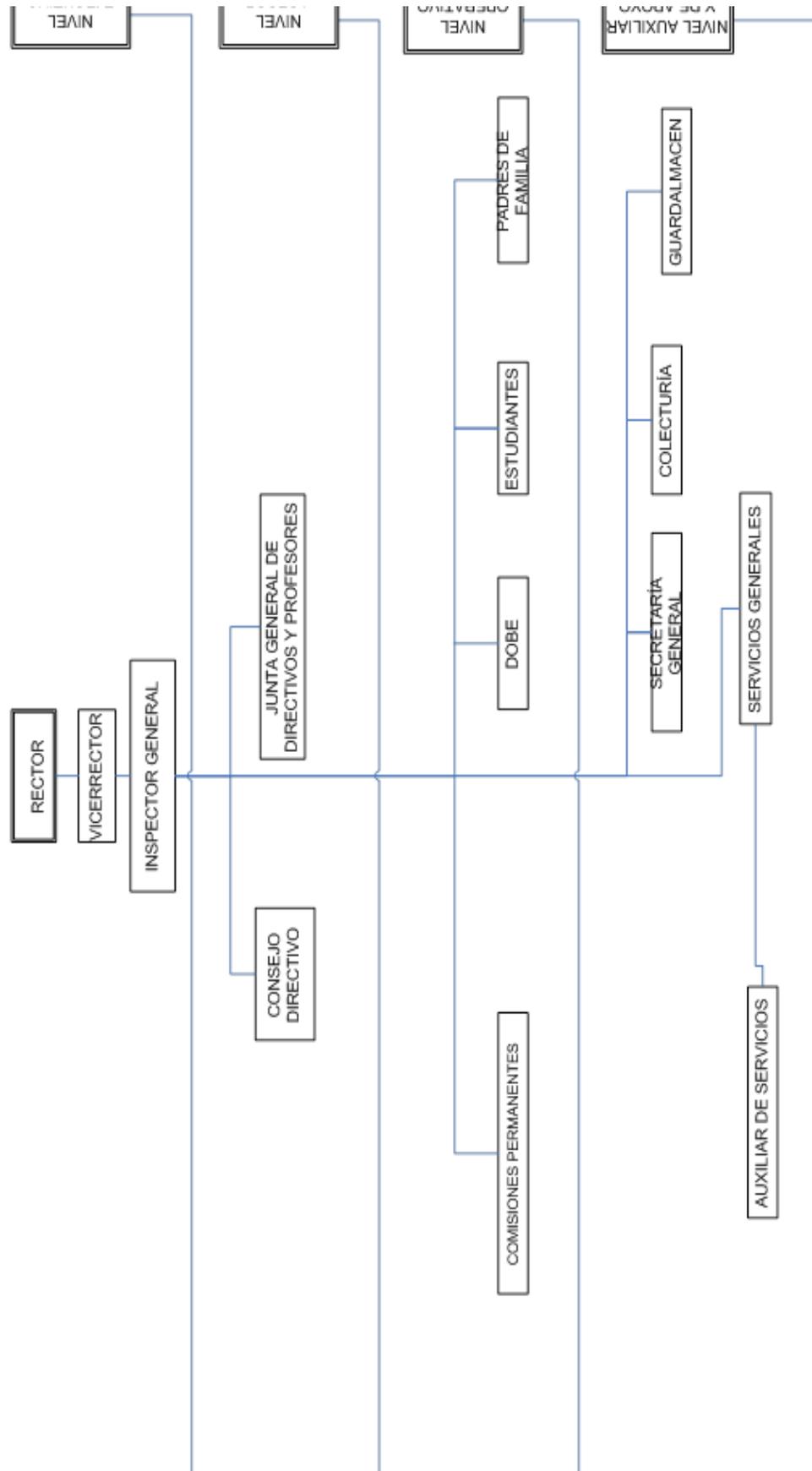


Anexo N°2

Diagrama Organizacional Del Colegio “Dr. M.B.C.M”



ORGANIGRAMA GENERAL DEL COLEGIO "DR. MANUEL BENJAMÍN CARRIÓN MORA"





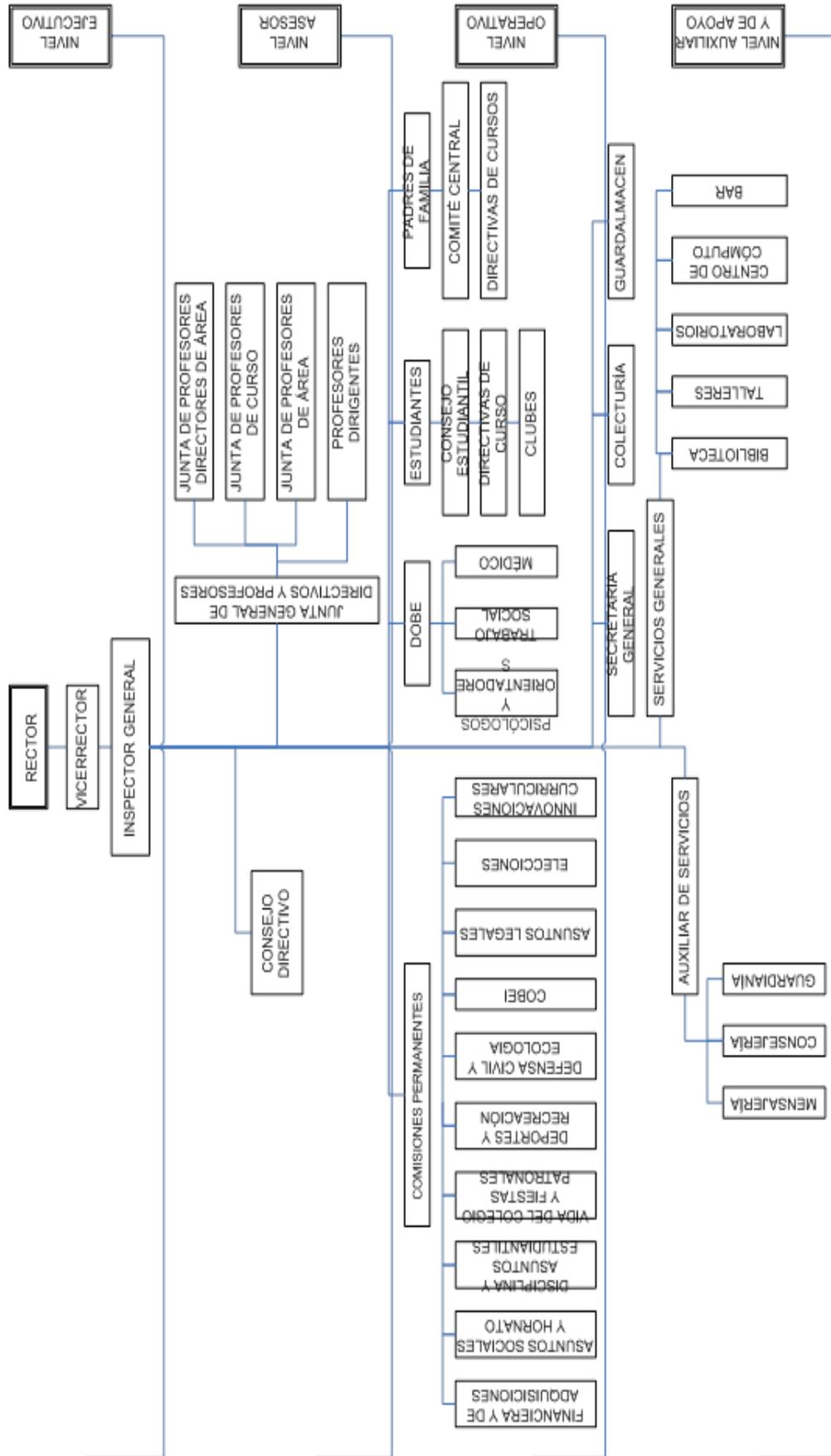
Anexo N°3

Diagrama Orgánico Funcional del Colegio “Dr. M.B.C.M”.



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

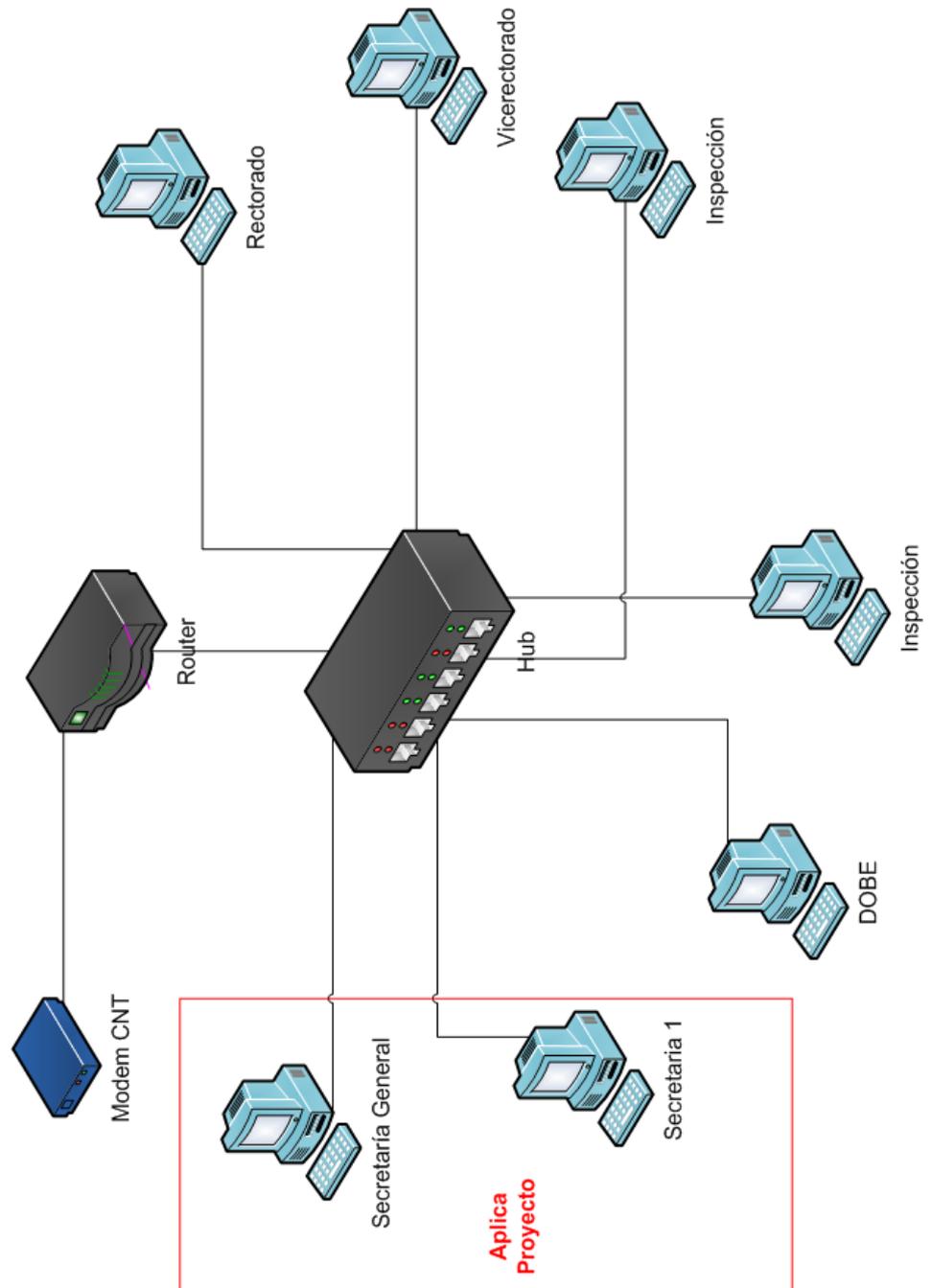
ORGANIGRAMA FUNCIONAL DEL COLEGIO "DR. MANUEL BENJAMÍN CARRIÓN MORA"





Anexo N°4

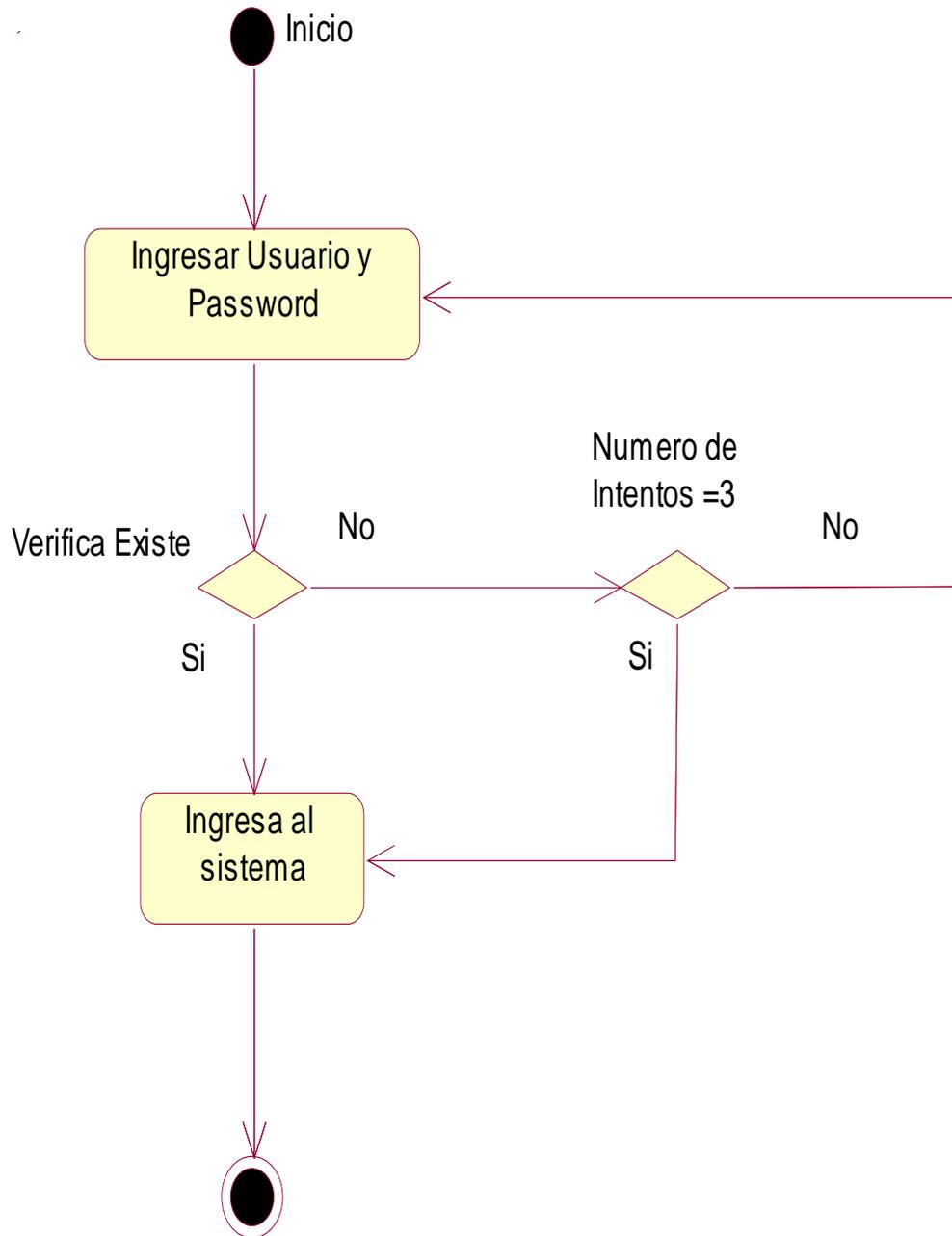
Diagrama De Red Del Colegio “Dr. M.B.C.M”





Anexo N°5

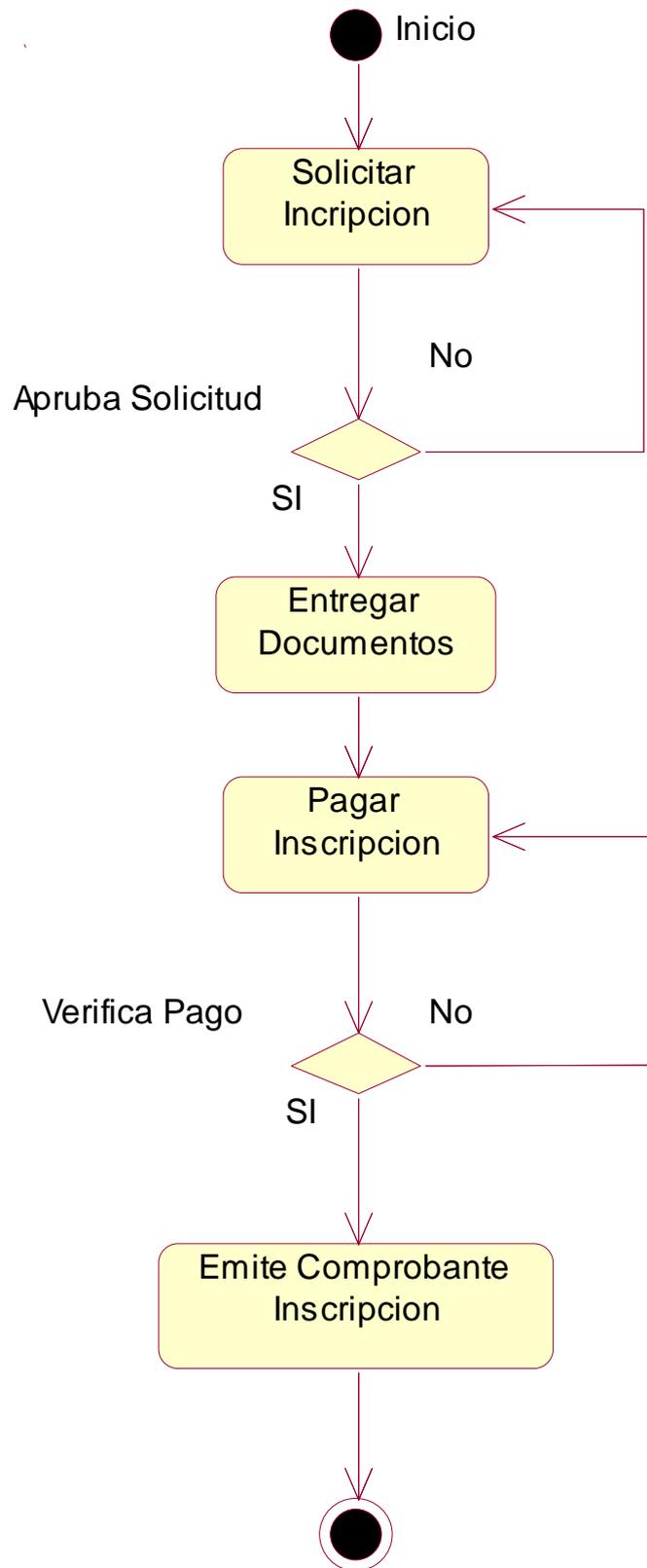
Diagrama De Estado De Flujo De Login





Anexo N°6

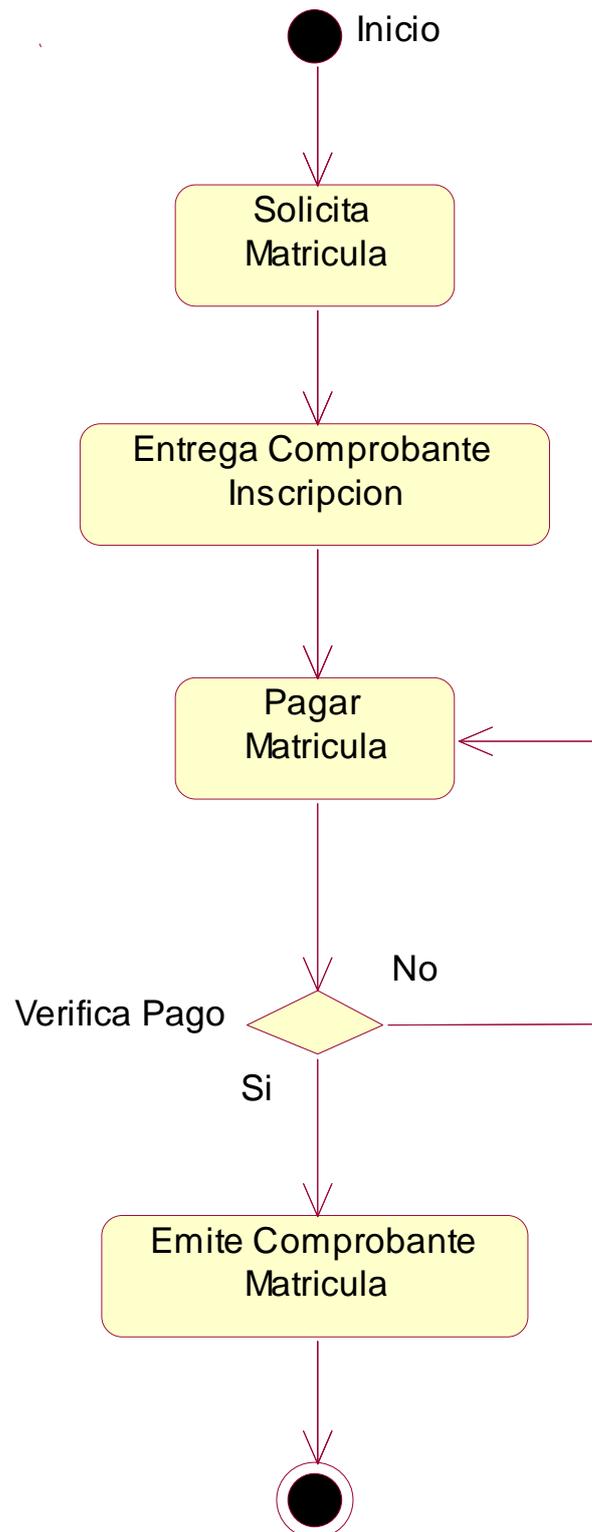
Diagrama De Estado De Flujo De Inscripción





Anexo N°7

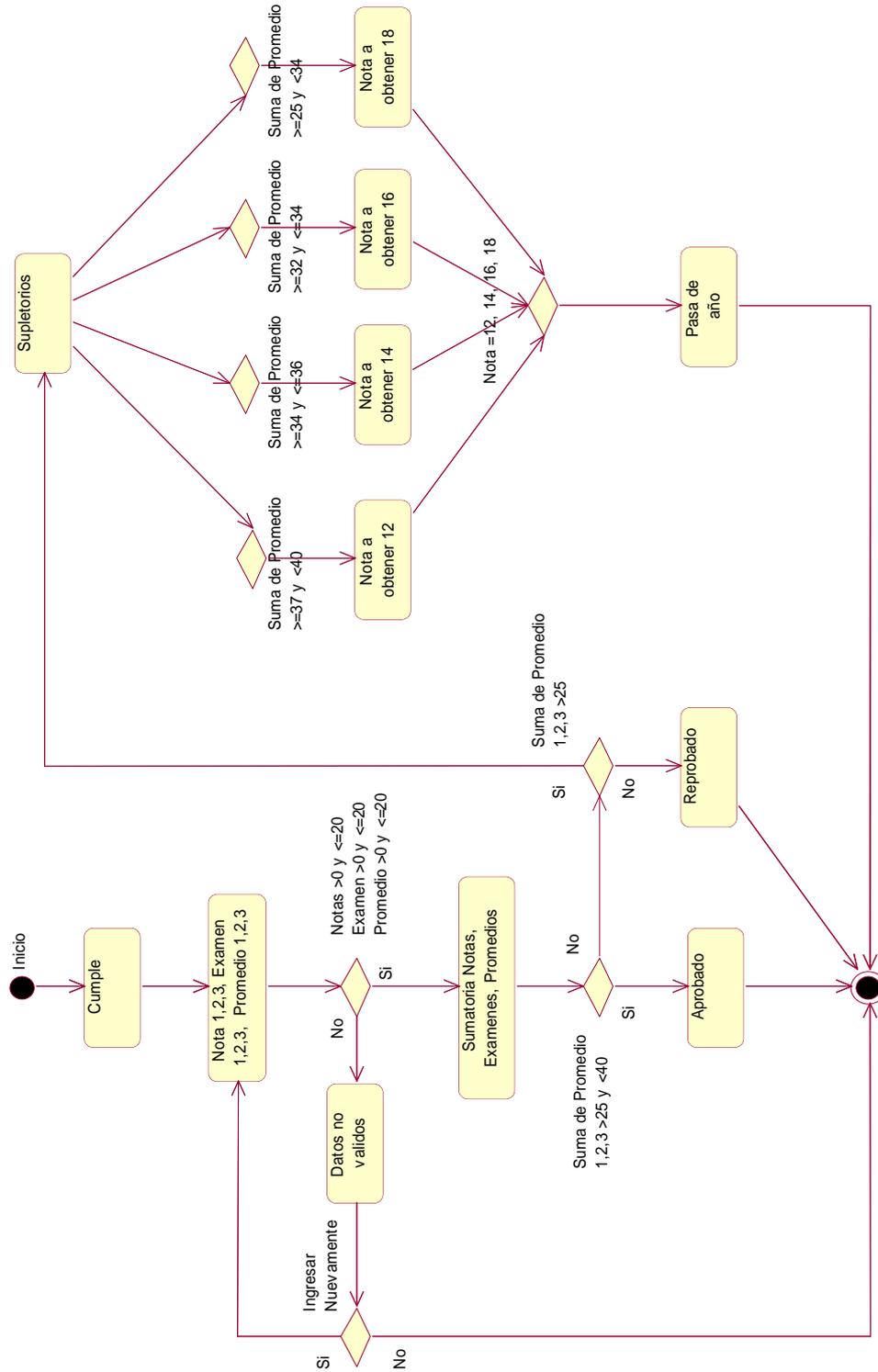
Diagrama De Estado De Flujo De Matriculación





Anexo N°8

Diagrama De Estado De Flujo De Notas





Anexo N°9

Documento SRS



Documento de Especificación de Requisitos de Usuario/Software

**“DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE CONTROL
ACADÉMICO (SISCONACA) PARA EL COLEGIO “DR MANUEL
BENJAMÍN CARRIÓN MORA”**

Fecha: 20 Enero del 2012

Versión: 1



Historia del Documento

Versión	Fecha	Razón del Cambio	Autor(es)
1.0	20/01/2012	Versión inicial.	Daniel Granizo



DESCRIPCIÓN GENERAL

La base de toda institución educativa son los estudiantes ya que si ellos no existieran de ahí nace la necesidad de la realización de un sistema que controle los procesos de gestión académica y llevar un correcto registro ello.

Para un manejo óptimo de la información se plantea la elección de una aplicación de un modelo que se concentrara en el correcto manejo de los procesos que realiza la institución.

El sistema funcionará reemplazando a la forma manual como se llevan actualmente la información los procesos de inscripciones y matriculaciones.

Para la realización de esta actividad se deberá estudiar la estructura definida.

Procesar con el software para obtener información requerida por el usuario.

Generará consultas y reportes de notas.

Siguiente trabajo definir el sistema para implementar en el colegio.

Plantear la interfaz de usuario que tendrá de acuerdo a los casos de uso levantados.

CARACTERÍSTICAS DE LOS USUARIOS

Los usuarios del sistema "SISCONACA" que interactúan directamente serán: el administrador, Digitadores de ingreso de información y digitador de ingreso de notas con sus respectivos permisos.

Tipo de usuario	Descripción
Usuario Administrador	Es el usuario que ingresa y da mantenimiento a todos los datos. Tiene un seguimiento de actividades auditable.
Usuario Digitador Ingresos Información	El usuario ingresos puede ingresar datos referentes a alumnos, profesores. No puede realizar cambios solo ingresos.
Usuario Digitador Inscripciones	El usuario inscripciones puede ingresar datos referentes a las inscripciones. No puede realizar cambios solo ingresos de inscripciones.
Usuario Digitador Matriculas	El usuario matriculas puede ingresar datos referentes a las matriculas del alumno. No puede realizar cambios solo ingresos.



PERSPECTIVA DEL PRODUCTO SEGÚN LOS USUARIOS/ CLIENTES

Cada usuario deberá ingresar con su respectiva contraseña con la finalidad de no afectar la información almacenada en el sistema.

Los usuarios esperan poder utilizar la aplicación conectados a la red obteniendo así procesos y reportes a su necesidad.

AMBIENTE OPERACIONAL DE LA SOLUCIÓN

El producto (software) se ejecutará bajo la plataforma de Windows puesto que todo el proceso de desarrollo y las respectivas pruebas, serán efectuados bajo este sistema operativo.

RELACIÓN CON OTROS PROYECTOS

Este proyecto se implantará completamente en la institución ya que no cuenta con ningún sistema.

La interacción con los datos almacenados en hojas de papel y algunos en Excel se limitará únicamente a consultas para ingresar al nuevo sistema.

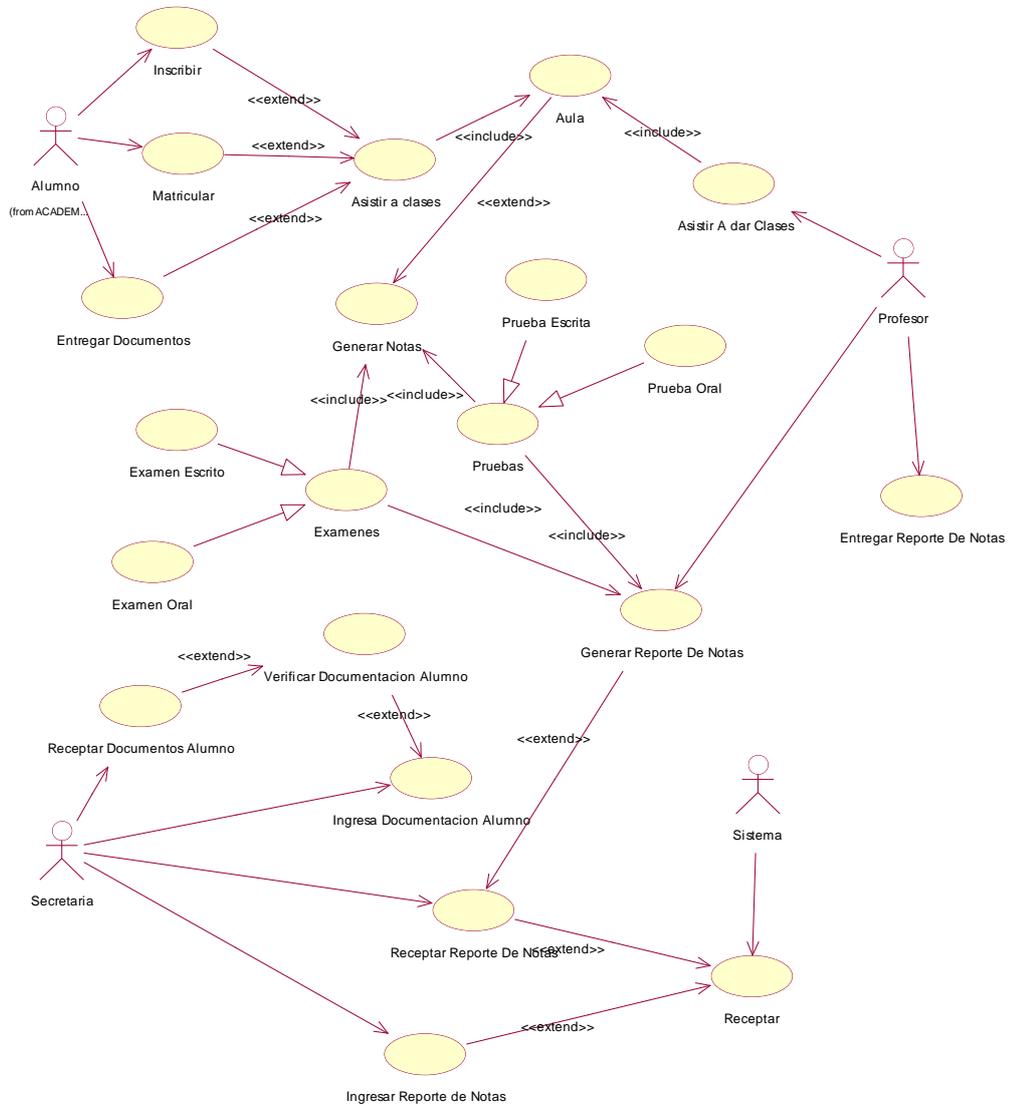
DESCRIPCIÓN DEL MODELO

La capa de presentación será desarrollada de acuerdo a algunas ideas aportadas por el personal de la institución en este caso las secretarias, pero con un diseño propio del software (SISCONACA).

El sistema constará de distintos procesos y reportes, que se acoplen a los requerimientos de los usuarios.



CASOS DE USO



Diccionario de Actores

Actor	Descripción
Secretaria	Es la persona que se encarga las inscripciones, matrículas y registro de notas.
Profesor	Es la persona de tomar pruebas, lecciones y exámenes a los alumnos y asentar notas.
Alumno	Persona responsable de generar notas mediante pruebas, lecciones y exámenes.



REQUERIMIENTOS DEL SISTEMA

REQUISITOS DE USUARIO

REQUISITOS DE CAPACIDADES

Identificador	RU0001
Nombre	Asignación de perfil para acceder a la red.
Descripción	Para poder acceder al sistema, el usuario debe poseer un nombre de usuario y clave, para realizar consultas o manipular los datos de manera restringida de acuerdo a su perfil de acceso.
Prioridad	Alta
Fuente	Tecnólogo Víctor Rodríguez
Estabilidad	Intrazable
Estado	Cumple
Listado de usuario	Todos
Entrega	Primera Iteración

REQUISITOS DE CALIDAD

Identificador	RU1001
Nombre	Modularidad
Descripción	El sistema está construido usando modularidad, ya que será utilizada por distintas maquinas en diferentes lugares.
Prioridad	Alta
Fuente	
Estabilidad	Intrazable
Estado	Cumple
Listado de usuario	-
Entrega	Primera Iteración



Identificador	RU1002
Nombre	Patrones de Diseño
Descripción	Este sistema tiene un diseño propio.
Prioridad	Medio
Fuente	Colegio
Estabilidad	Transable
Estado	Nuevo
Listado de usuario	-
Entrega	Primera Iteración

REQUISITOS DE RESTRICCIÓN

Identificador	RU2001
Nombre	Sistema utilizado bajo la plataforma Windows XP, 7 con Net. Compact Framework 4.0 SP2 para Laptops y PC
Descripción	El sistema usado trabaja en plataforma Windows XP con Net. Compact Framework 4.0 SP2 para Laptops y PC.
Prioridad	Alta
Fuente	Colegio
Estabilidad	Intrazable
Estado	Cumple
Listado de usuario	-
Entrega	Primera Iteración

REQUISITOS DE SOFTWARE

REQUISITOS FUNCIONALES



Identificador	RS0001
Nombre	Verificación Automática del Usuario
Descripción	Proceso para verificar al usuario en el sistema.
Prioridad	Alta
Fuente	Daniel Granizo
Estabilidad	Intrazable
Estado	Cumple
Listado de usuario	-
Entrega	Primera Iteración

Identificador	RS0002
Nombre	Ingreso inscripciones y matriculas
Descripción	Se registra en el sistema los datos personales del alumno y a que nivel va.
Prioridad	Alto.
Fuente	Colegio
Estabilidad	Transable
Estado	Cumple
Entrega	Primera Iteración

REQUISITOS DE INTERFAZ

Identificador	RS1001
Nombre	Implantar la Interfaz
Descripción	Diseño propio del sistema (SISCO).
Prioridad	Alta



Fuente	Daniel Granizo
Estabilidad	Intrazable
Estado	Cumple
Entrega	Primera Iteración

OPERACIONALES

Identificador	RS2001
Nombre	Mensajes de Alertas
Descripción	El usuario deberá corroborar las acciones que desea realizar, y la clase de reportes de notas que desea.
Prioridad	Alta
Fuente	Ing. Richard Espinosa
Estabilidad	Intrazable
Estado	Cumple
Entrega	Primera Iteración

MANTENIBILIDAD

Identificador	RS3001
Nombre	Código mantenible y expandible
Descripción	Esta funcionalidad permite a un futuro desarrollador del proyecto extender de forma rápida la composición del código de la aplicación, con el objetivo de realizar modificaciones a este. Se entiende por mantenible o expandible al código que se encuentre separado en capas y utilice correctamente modularidad y patrones de diseño.
Prioridad	Critico
Fuente	Daniel Granizo
Estabilidad	Transable



Estado	Nuevo
Entrega	Primera Iteración

MATRIZ DE TRAZADO: REQUISITOS DE USUARIO VS REQUISITOS DE SOFTWARE

	RS0001	RS0002	RS0003	RS1001	RS1002	RS2001	RS3001
RU0001	X					X	
RU1001	X			X		X	
RU1002		X			X		
RU2001				X			
RU2002				X			

PRUEBAS DEL SISTEMA

PRUEBAS DE USUARIO

ID	PR0001
Requisito de usuario asociado	RU0001
Nombre	Asignación de perfil para acceder al sistema
Descripción	El usuario podrá acceder al sistema mediante un nombre usuario y su contraseña caso contrario no podrá acceder.
Peor Valor Aceptable	No tener nombre de usuario y contraseña
Valor Planificado	Nombre usuario y contraseña
Tipo de usuario	Administrador, Usuario Ingresos, Usuario inscripciones, Usuario matriculas.
Estado (cumple / no cumple)	



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

ID	PR0002
Requisito de usuario asociado	RU1001
Nombre	Modularidad
Descripción	Se realizara una prueba de estrés para ver cuántos usuarios pueden estar conectados al mismo tiempo y con esto saber si el sistema colapsa o no.
Peor Valor Aceptable	Colapsar el sistema
Valor Planificado	No colapse el sistema estando conectados 5 usuarios.
Tipo de usuario	Todos
Estado (cumple / no cumple)	

ID	PR0003
Requisito de usuario asociado	RU1002
Nombre	Patrones de diseño.
Descripción	Se utilizara un diseño propio del sistema (SISCOA) con SQLSEVER EXPRESS 2008, lenguaje de programación Visual C# 2008 con las cuales esta creado el sistema.
Peor Valor Aceptable	Ninguno
Valor Planificado	Tener un de diseño de calidad.
Tipo de usuario	Todos
Estado (cumple / no cumple)	

ID	PR0004
Requisito de usuario asociado	RU2001
Nombre	Compatibilidad de plataforma
Descripción	Se verificará que el sistema sea compatible con la plataforma con



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

	la cual cuenta la institución esto se realizará, corriendo la aplicación en dicha plataforma.
Peor Valor Aceptable	No ser compatible
Valor Planificado	Correr la aplicación en la plataforma de la institución
Tipo de usuario	Todos
Estado (cumple / no cumple)	

ID	PR0005
Requisito de usuario asociado	RU2002
Nombre	Utilización de protocolos.
Descripción	El sistema deberá estar conectado al servidor de base de datos.
Peor Valor Aceptable	No conectarse al servidor.
Valor Planificado	Estar el sistema siempre conectado al servidor.
Tipo de usuario	Todos
Estado (cumple / no cumple)	

ID	PR0006
Requisito de usuario asociado	RS0001
Nombre	Verificación de Usuario
Descripción	El usuario deberá ingresar su nombre o nick y password el cual será validado con la base de datos, así podrá tener acceso al modulo
Peor Valor Aceptable	No podrá ingresar al sistema
Valor Planificado	Podrá ingresar a la aplicación
Tipo de usuario	Todos



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

Estado (cumple / no cumple)	
-----------------------------	--

ID	PR0008
Requisito de usuario asociado	RS0002
Nombre	Nombre de usuario activo en el modulo
Descripción	El usuario ingresado en el módulo, tendrá en la parte superior izquierda su nick o nombre como usuario activado
Peor Valor Aceptable	No tendrá su nick o nombre activado
Valor Planificado	Se visualizara su nick o nombre en la parte superior izquierda.
Tipo de usuario	Todos
Estado (cumple / no cumple)	

ID	PR0009
Requisito de usuario asociado	RS1001
Nombre	Prueba de eliminación de usuarios en el sistema.
Descripción	Se podrá deshabilitar a usuarios
Peor Valor Aceptable	Que siga constando su nick o nombre de usuario
Valor Planificado	Al eliminar bloquear al usuario.
Tipo de usuario	Todos
Estado (cumple / no cumple)	



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

ID	PR0010
Requisito de usuario asociado	RS1002
Nombre	Prueba de conexión de usuarios a una sesión
Descripción	En un número de usuarios creados, tendrá visible su nick como sección activa.
Peor Valor Aceptable	No se visualizara el usuario
Valor Planificado	Se visualizara al usuario.
Tipo de usuario	Todos
Estado (cumple / no cumple)	

ID	PR0011
Requisito de usuario asociado	RS2001
Nombre	Prueba de despliegue de mensaje informativo corroborando que la información es la correcta
Descripción	Cuando se realiza una consulta a la base de datos con parámetros en específico el sistema debe mostrar la información correcta
Peor Valor Aceptable	No visualizar el mensaje
Valor Planificado	Visualizar el mensaje
Tipo de usuario	Todos
Estado (cumple / no cumple)	



MATRIZ DE TRAZADO REQUISITOS DE USUARIO VS. PRUEBAS

	RU0001	RU1001	RU1002	RU2001	RU2002
PR0001	X				
PR0002		X			
PR0003			X		
PR0004				X	
PR0005					X
PR0006					
PR0007					
PR0008					
PR0009					
PR0010					
PR0011					
PR0012					



Anexo N°10

Modelo Físico



Anexo N°11

Diccionario de Datos



Schema	Table
dbo	Tbl Alumno
dbo	Tbl Asistencias
dbo	Tbl Auditoria
dbo	Tbl Detalle Profesor
dbo	Tbl DTNotasAño
dbo	Tbl DTNotasSemestre
dbo	Tbl Especialidades
dbo	Tbl Faltas
dbo	Tbl Inscripciones
dbo	Tbl Jornadas
dbo	Tbl Materias
dbo	Tbl MateriaXAlumno
dbo	Tbl Matriculas
dbo	Tbl Niveles
dbo	Tbl Notas
dbo	Tbl Parametros
dbo	Tbl Periodos
dbo	Tbl Profesor
dbo	Tbl TipoMatricula
dbo	Tbl Usuario

[dbo].[Tbl_Alumno]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Alu_Codigo	Int	4	N	Y	N	
Usu_Codigo	Int	4	N	Y	Y	
Alu_Representante	VarChar (50)	50	N	N	N	
Alu_Recorrido	Bit	1	Y	N	N	
Alu_PagoRecorrido	Decimal (4,2)	5	Y	N	N	
Alu_Estado	Char (1)	1	Y	N	N	

[dbo].[Tbl_Asistencias]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Asis_Codigo	Int	4	N	Y	N	
Asis_Fecha	DateTime	8	Y	N	N	
Asis_Estado	Char (1)	1	Y	N	N	
Alu_Codigo	Int	4	N	N	Y	
Falt_Codigo	Int	4	N	N	Y	



Usu_Codigo	Int	4	N	N	Y	
------------	-----	---	---	---	---	--

[dbo].[Tbl_Auditoria]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Aud_Codigo	Int	4	N	Y	N	
Aud_Descripcion	VarChar (30)	30	N	N	N	
Aud_Accion	VarChar (30)	30	N	N	N	
Aud_Fecha	DateTime	8	N	N	N	
Usu_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Detalle_Profesor]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
DP_Codigo	Char (1)	1	N	Y	N	
Prof_Codigo	Int	4	N	N	Y	
Period_Codigo	Int	4	N	N	Y	
Jor_Codigo	Int	4	N	N	Y	
Espe_Codigo	Int	4	N	N	Y	
Niv_Codigo	Int	4	N	N	Y	
Niv_Paralelo	Char (1)	1	N	N	Y	
Mate_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_DTNotasAño]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
DTNA_Codigo	Int	4	N	Y	N	
Notas_Codigo	Int	4	N	N	Y	
Mate_Codigo	Int	4	N	N	Y	
DTNA_Nota1	Decimal (4,2)	5	Y	N	N	
DTNA_Nota2	Decimal (4,2)	5	Y	N	N	
DTNA_Nota3	Decimal (4,2)	5	Y	N	N	
DTNA_Examen1	Decimal (4,2)	5	Y	N	N	
DTNA_Promedio1	Decimal (4,2)	5	Y	N	N	
DTNA_Nota4	Decimal (4,2)	5	Y	N	N	
DTNA_Nota5	Decimal (4,2)	5	Y	N	N	
DTNA_Nota6	Decimal (4,2)	5	Y	N	N	
DTNA_Examen2	Decimal (4,2)	5	Y	N	N	
DTNA_Promedio2	Decimal (4,2)	5	Y	N	N	
DTNA_Nota7	Decimal (4,2)	5	Y	N	N	
DTNA_Nota8	Decimal (4,2)	5	Y	N	N	



DTNA_Nota9	Decimal (4,2)	5	Y	N	N	
DTNA_Promedio3	Decimal (4,2)	5	Y	N	N	
DTNA_Examen3	Decimal (4,2)	5	Y	N	N	
DTNA_Sumatoria	Decimal (5,2)	5	Y	N	N	
DTNA_PromedioG	Decimal (4,2)	5	Y	N	N	
DTNA_Observacion	VarChar (60)	60	Y	N	N	

[dbo].[Tbl_DTNotasSemestre]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
DTNS_Codigo	Int	4	N	Y	N	
Mate_Codigo	Int	4	N	N	Y	
DTNS_Bimes1	Decimal (4,2)	5	Y	N	N	
DTNS_Bimes2	Decimal (4,2)	5	Y	N	N	
DTNS_Bimes3	Decimal (4,2)	5	Y	N	N	
DTNS_Sumatoria	Decimal (5,2)	5	Y	N	N	
DTNS_Promedio	Decimal (4,2)	5	Y	N	N	
DTNS_Observacion	VarChar (60)	60	Y	N	N	
Notas_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Especialidades]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Espe_Codigo	Int	4	N	Y	N	
Espe_Nombre	VarChar (50)	50	Y	N	N	
Espe_Estado	Char (1)	1	Y	N	N	
Jor_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Faltas]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Falt_Codigo	Int	4	N	Y	N	
Falt_Nombres	VarChar (50)	50	Y	N	N	
Falt_Fecha	DateTime	8	Y	N	N	
Falt_Estado	Char (1)	1	N	N	N	

[dbo].[Tbl_Inscripciones]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Ins_Codigo	Int	4	N	Y	N	
Ins_Fecha	DateTime	8	Y	N	N	
ins_PagoTotal	Decimal (4,2)	5	Y	N	N	



Ins_Saldo	Decimal (4,2)	5	Y	N	N	
Ins_Descripcion	VarChar (100)	100	Y	N	N	
Alu_Codigo	Int	4	N	N	Y	
Usu_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Jornadas]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Jor_Codigo	Int	4	N	Y	N	
Jor_Nombre	VarChar (60)	60	Y	N	N	
Jor_Estado	Char (1)	1	Y	N	N	
Period_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Materias]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Mate_Codigo	Int	4	N	Y	N	
Mate_Nombre	VarChar (60)	60	Y	N	N	
Mate_Estado	Char (1)	1	Y	N	N	
Niv_Codigo	Int	4	N	N	Y	
Niv_Paralelo	Char (1)	1	N	N	Y	

[dbo].[Tbl_MateriaXAlumno]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Alu_Codigo	Int	4	N	N	Y	
Mate_Codigo	Int	4	N	N	Y	
MXA_Codigo	Int	4	N	Y	N	
MateXAlu_Estado	Char (1)	1	N	N	N	
Usu_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Matriculas]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Matri_Codigo	Int	4	N	Y	N	
Jor_Codigo	Int	4	N	N	Y	
TM_Codigo	Int	4	N	N	Y	
Period_Codigo	Int	4	N	N	Y	
Niv_Codigo	Int	4	N	N	Y	
Niv_Paralelo	Char (1)	1	N	N	Y	
Espe_Codigo	Int	4	N	N	Y	
Matri_Fecha	DateTime	8	Y	N	N	



Matri_Pago	Decimal (5,2)	5	Y	N	N	
Matri_Estado	Char (1)	1	Y	N	N	
Alu_Codigo	Int	4	N	N	Y	
Usu_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Niveles]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Niv_Codigo	Int	4	N	Y	N	
Niv_Nombre	VarChar (60)	60	Y	N	N	
Niv_Estado	Char (1)	1	Y	N	N	
Niv_Paralelo	Char (1)	1	N	Y	N	
Espe_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Notas]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Notas_Codigo	Int	4	N	Y	N	
Notas_Aprobacion	Char (1)	1	N	N	N	
Matri_Codigo	Int	4	N	N	Y	

[dbo].[Tbl_Parametros]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
PRM_Codigo	Int	4	N	Y	N	
PRM_Id_Padre	Int	4	N	N	N	
PRM_Id_hijo	Int	4	N	N	N	
PRM_Tipo_Usuario	VarChar (20)	20	N	N	N	
PRM_Descripcion	VarChar (20)	20	Y	N	N	
PRM_Estado	Char (1)	1	N	N	N	

[dbo].[Tbl_Periodos]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Period_Codigo	Int	4	N	Y	N	
Period_Nombre	VarChar (50)	50	Y	N	N	
Period_Estado	Char (1)	1	Y	N	N	
Period_Descripcion	VarChar (100)	100	Y	N	N	
Period_Fechalni	DateTime	8	Y	N	N	
Period_FechaFin	DateTime	8	Y	N	N	

[dbo].[Tbl_Profesor]:



Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Prof_Codigo	Int	4	N	Y	N	
Usu_Codigo	Int	4	N	N	Y	
Prof_Titulo	VarChar (20)	20	Y	N	N	
Prof_Estado	Char (1)	1	Y	N	N	

[dbo].[Tbl_TipoMatricula]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
TM_Codigo	Int	4	N	Y	N	
TM_Nombre	VarChar (60)	60	Y	N	N	
TM_Recarga	Decimal (4,2)	5	Y	N	N	
Tm_Estado	Char (1)	1	Y	N	N	

[dbo].[Tbl_Usuario]:

Column	Datatype	Size	Nullable	InPrimaryKey	IsForeignKey	Description
Usu_Codigo	Int	4	N	Y	N	
Usu_Cedula	VarChar (10)	10	Y	N	N	
Usu_Nombre	VarChar (50)	50	Y	N	N	
Usu_Apellidos	VarChar (60)	60	Y	N	N	
Usu_Telefono	VarChar (10)	10	Y	N	N	
Usu_Genero	Char (1)	1	Y	N	N	
Usu_Direccion	VarChar (100)	100	Y	N	N	
Usu_Estado	Char (1)	1	Y	N	N	
Usu_Foto	Text	16	Y	N	N	
PRM_Codigo	Int	4	N	N	Y	



Anexo N°12

Documento SDS



Documento de Diseño (SDS)

“DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE CONTROL ACADÉMICO PARA EL COLEGIO DR”MANUEL BENJAMÍN CARRIÓN MORA”

Fecha: 15 Marzo del 2012

Versión: 1.0



Historia del Documento

Versión	Fecha	Razón del Cambio	Autor(es)
1.0	15/03/2012	Versión inicial.	Daniel Granizo



DESCRIPCIÓN GENERAL

En el ingreso al sistema, el usuario tendrá acceso único y exclusivamente con sus respectivas identificaciones, logrando así obtener los resultados requeridos.

El primer paso es realizar el análisis de los requerimientos y el levantamiento de los mismos.

El siguiente es definir las clases para cada uno de los paquetes de información de gestión académica, analizarlos y organizarlos aplicando los conocimientos de programación orientada a objetos e indicando cada uno de sus estructuras y relaciones.

A continuación se indica cada una de las funciones del sistema:

- Proponer la interfaz de usuario que tendrá de acuerdo a los casos de uso levantados.
- Llevar un control oportuno, en el ingreso de los procesos académicos.
- Visualizar una interfaz, que se presenta de forma amigable al usuario.
- Se procesará con el software SISCONACA para obtener información requerida por el usuario.

SERVICIOS A BRINDAR POR EL PRODUCTO

El sistema concluido ofrecerá los siguientes servicios:

- Permitirá realizar consultas por perfil de usuario.
- Generará reportes de las calificaciones de los alumnos, listados de alumnos, mismas que se presentan en una pantalla amigable al usuario.
- Adicionalmente mostrará reportes generales listado por materia, profesor, curso o nivel, especialidad.
- Permitirá realizar actualizaciones de información.
- Proveerá al administrador permisos.

DISEÑO ARQUITECTÓNICO

ARQUITECTURA FÍSICA

ESQUEMA GENERAL DEL SISTEMA

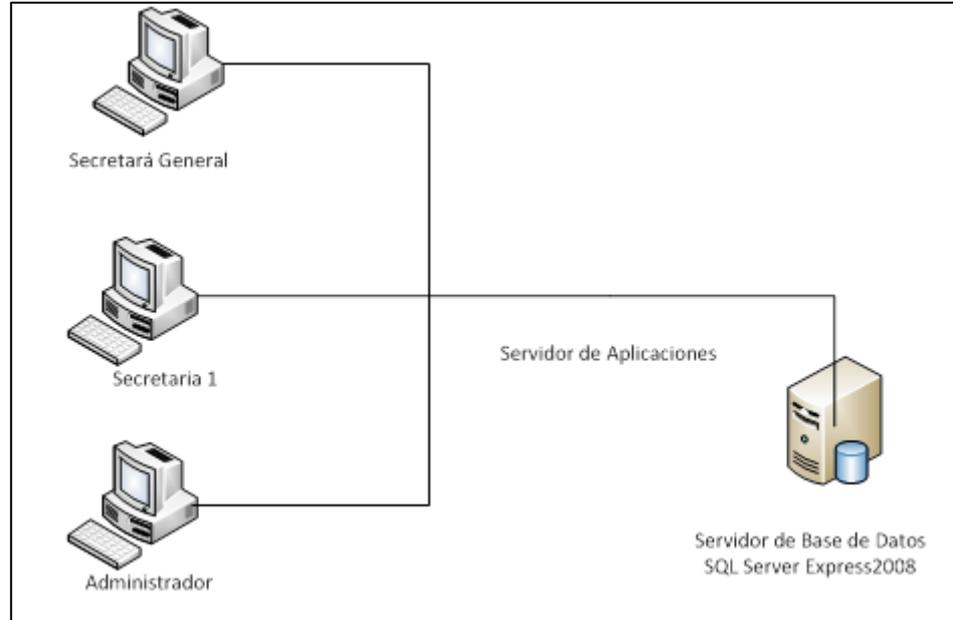
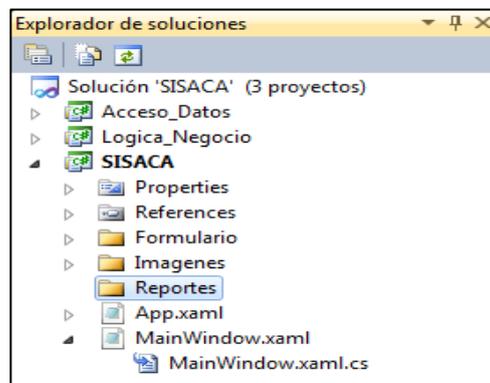


Gráfico Nº 7: Esquema general de la arquitectura física del sistema

Capa Interfaz (Cliente)

Es la capa de presentación de un computador personal cualquiera provisto de un enlace con conexión al servidor.

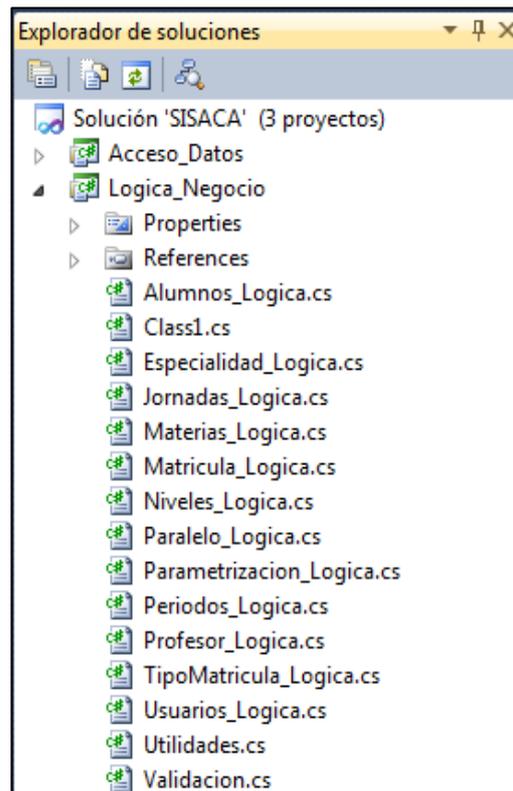
Permite al usuario mediante la red LAN conectarse con el servidor, y de esta manera realizar las consultas requeridas por el usuario.





Capa Lógica de Negocio (Servidor Aplicaciones)

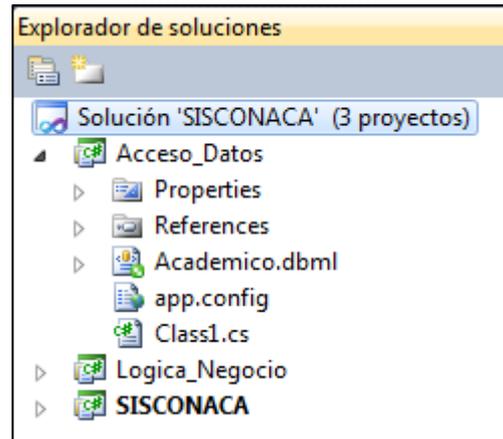
El servidor de aplicaciones responde a las peticiones de los usuarios, maneja las transacciones según la lógica del sistema y contacta al servidor de base de datos a través de consultas SQL. Es capaz de generar respuestas en la forma de textos. Es la que entrega resultados de los requerimientos realizados por el usuario, a la vez que se conecta con el dominio del servidor, brindando un servicio rápido y eficiente.



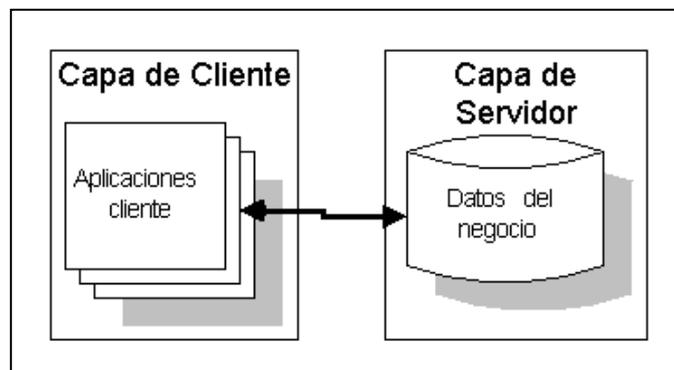
Capa de datos (Servidor de Base de Datos)

Se encuentran la base de datos del sistema donde obtendrá y añadirá los datos entregando respuestas al servidor de base de datos.

El servidor de BDD está encargado de guardar la información sobre la que se generan los distintos procesos del sistema, utilizando el motor de base de datos SQLServer 2008 Express, respondiendo, sólo a las peticiones del Servidor de Aplicaciones.



ESQUEMA ESPECÍFICO DEL SISTEMA



El cliente, a través de un computador, realiza consultas al servidor de base de datos, el cual a su vez ejecuta la acción requerida a través de sus controladores, quienes contienen la lógica requerida, extrayendo datos de la BD retornando la petición solicitada por el cliente a través del servidor de aplicaciones.

De acuerdo con la solución ya procesada, el controlador envía esta información a las vistas correspondientes del servidor de aplicaciones entregando los datos requeridos como del alumno, docente, notas del alumnado de acuerdo a la lista de los que ofrece el sistema en el formato (.docx) o solo la interfaz en un formulario.

Arquitectura Lógica

Diagrama General

Conceptualmente, el sistema SISCONACA (incluyendo su panel de actualización y administración) se puede dividir:

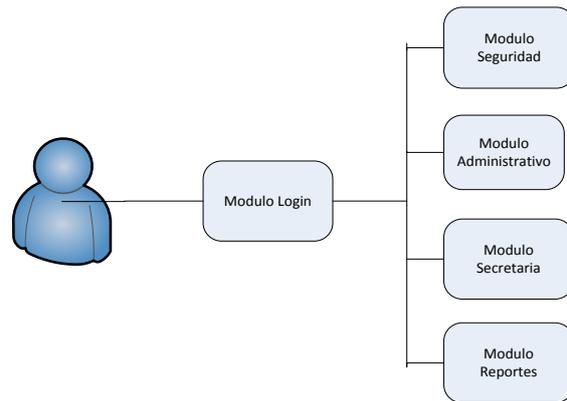


Gráfico N° 9: Diagrama general de la arquitectura lógica

Bajo este esquema, el usuario puede acceder solo al módulo al que este asignado, y su filtro será el módulo de Login del sistema

Dependiendo del tipo de usuario, se reconocen las siguientes acciones posibles:

Usuario Secretaria

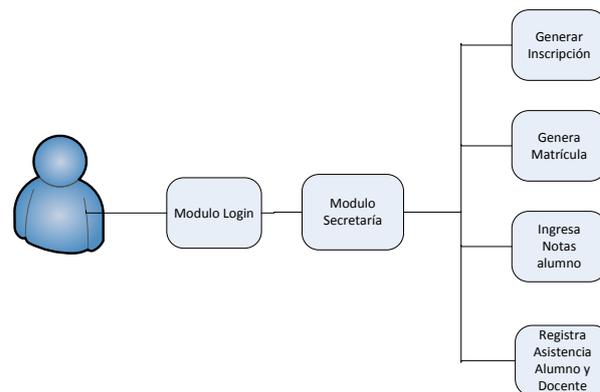


Gráfico N° 10: Acciones posibles del usuario Secretaria

Usuario Docente

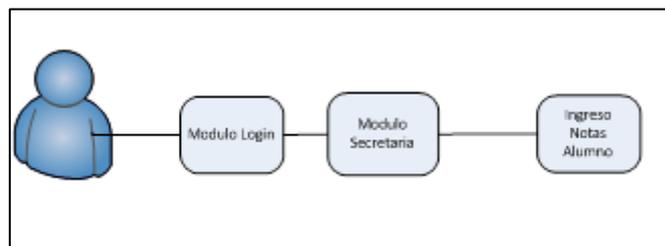


Gráfico N° 11: Acciones posibles del usuario Docente

Usuario Administrador

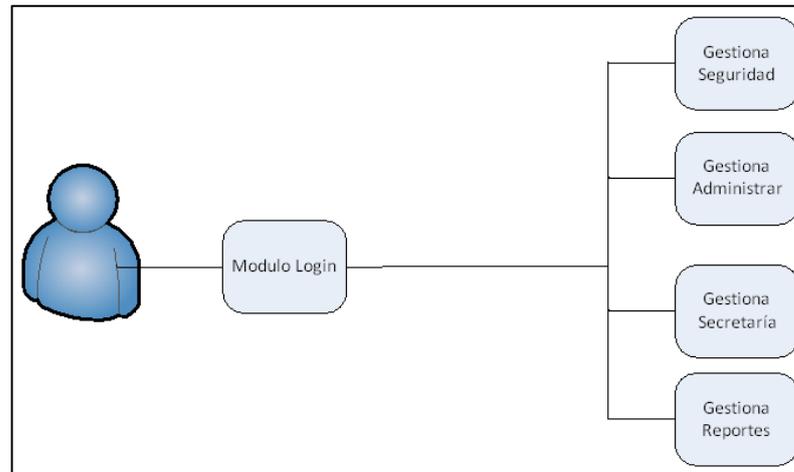
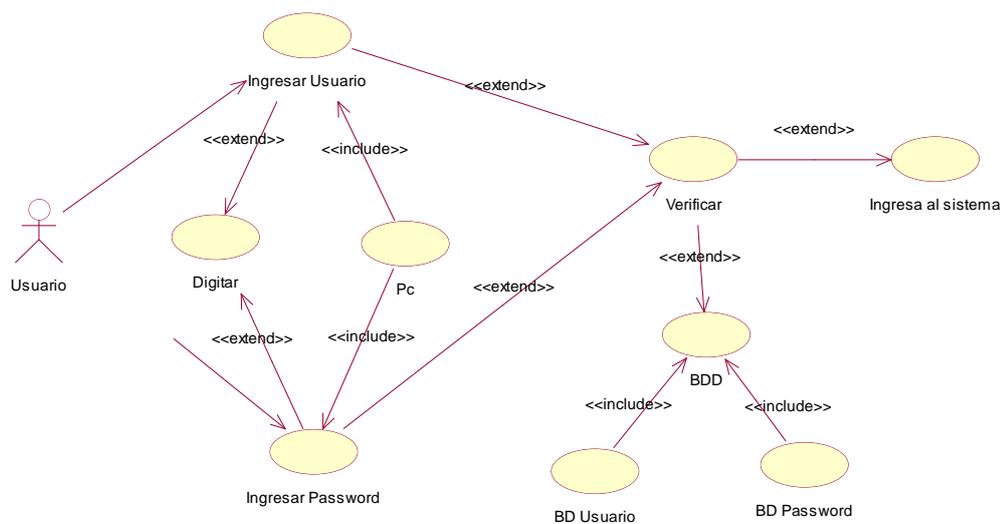


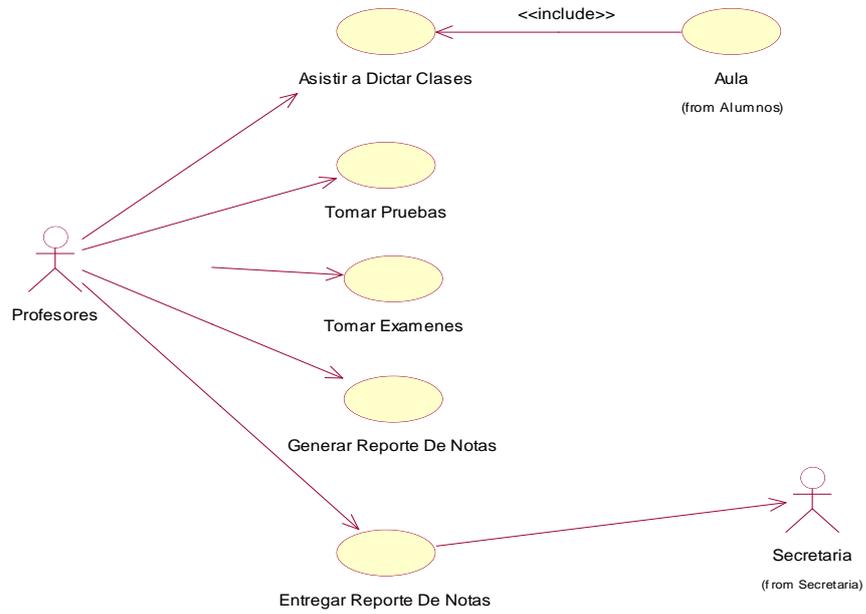
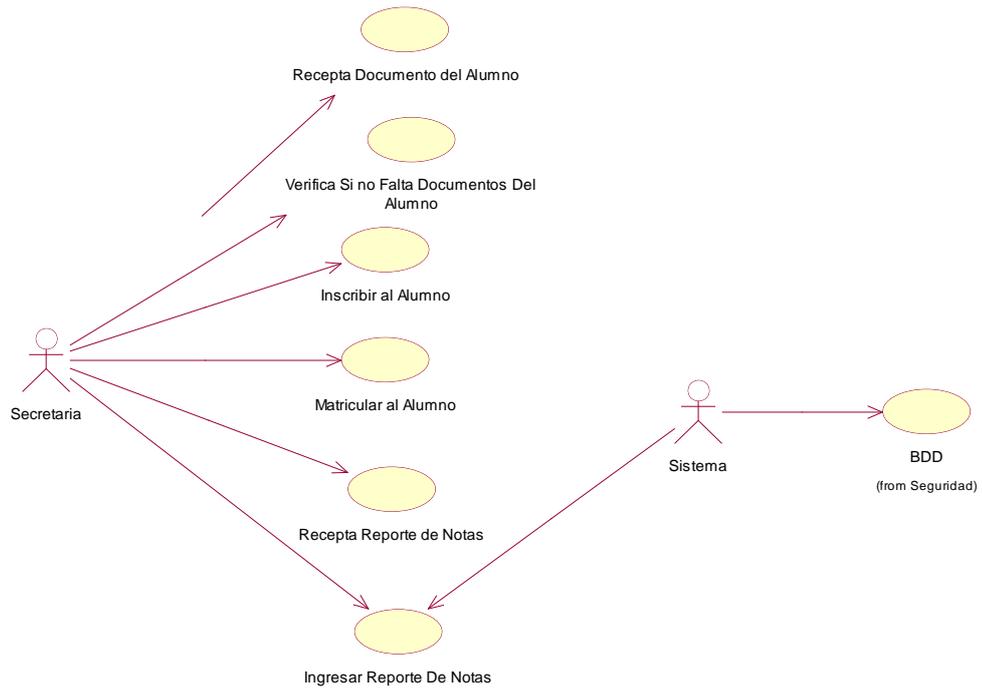
Gráfico N° 13: Acciones posibles del usuario administrador

Modelo de datos





INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA



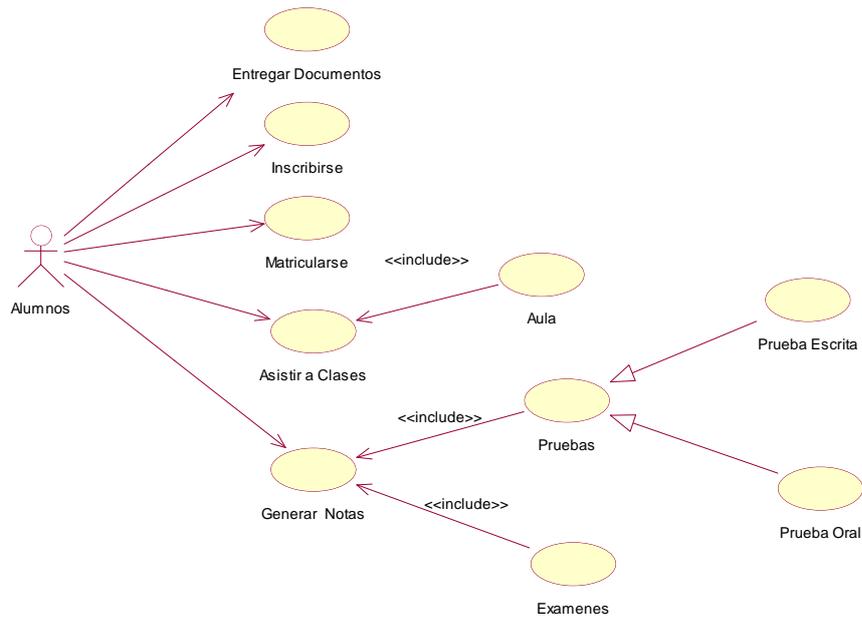
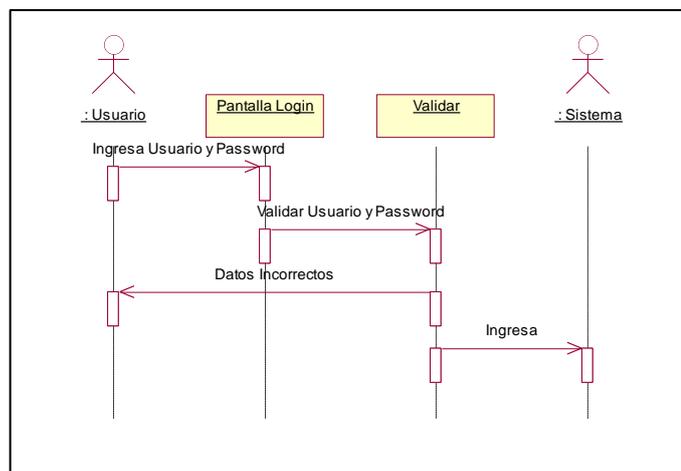


Diagrama de Secuencia

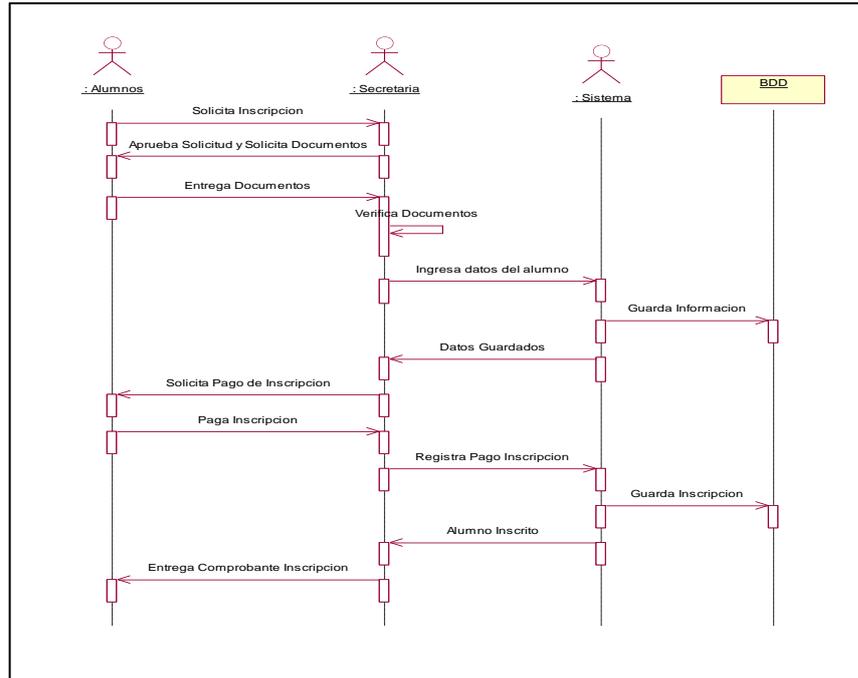
En los diagramas de secuencia se visualizaran todos los pasos a seguir, a partir de un actor específico se irá demostrando por donde el usuario del sistema tiene que pasar y dependiendo del módulo debe ingresar los datos para respectiva validación de los mismos.

El diagrama de secuencia son los caminos por donde el usuario debe validar para llegar al proceso específico.

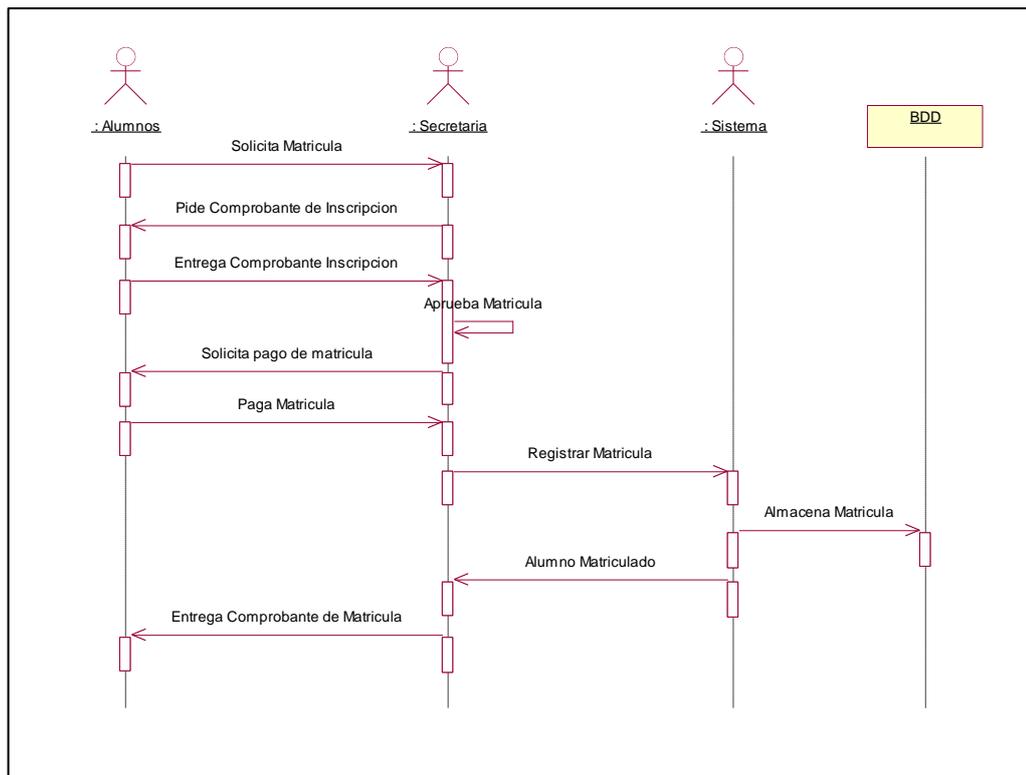
Ds_01 Ingreso al Sistema



Ds_02 Inscripción

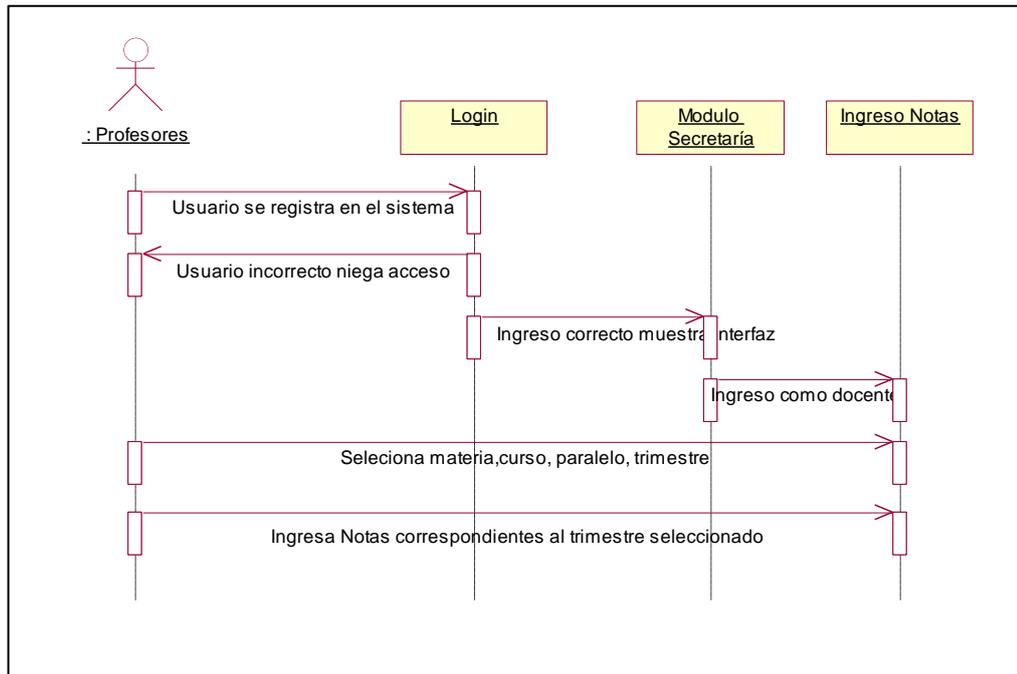


Ds_03 Matriculación

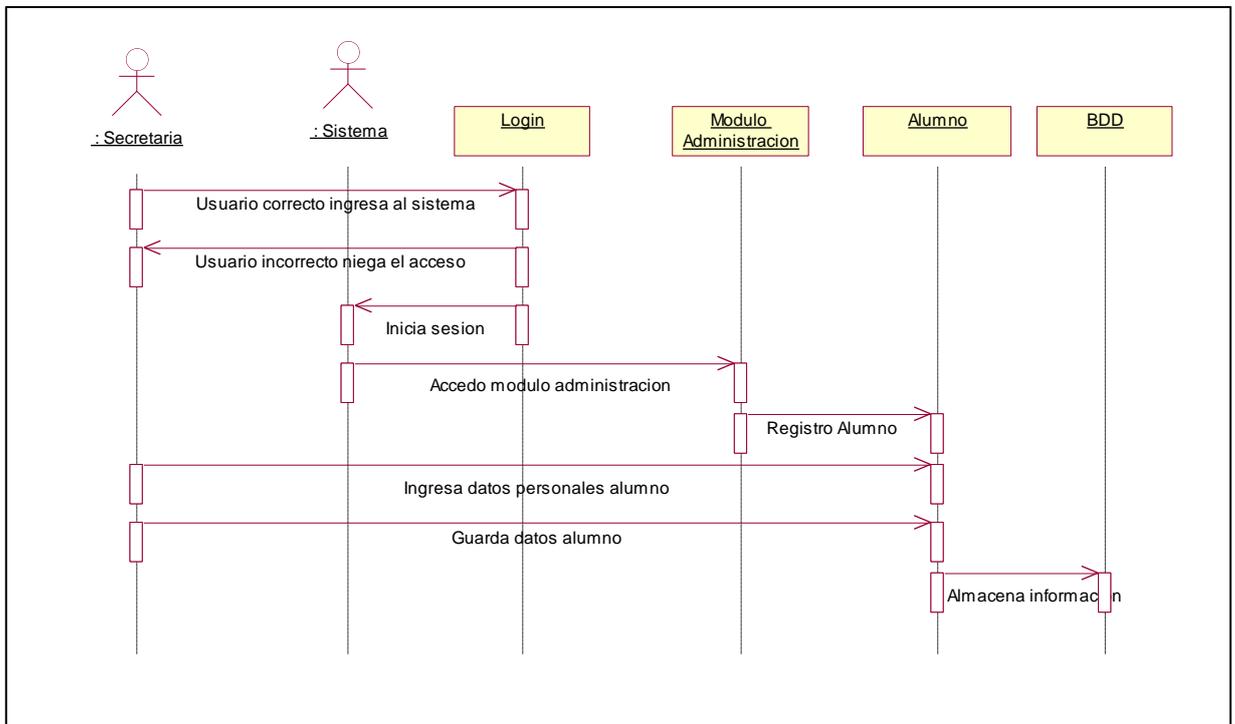




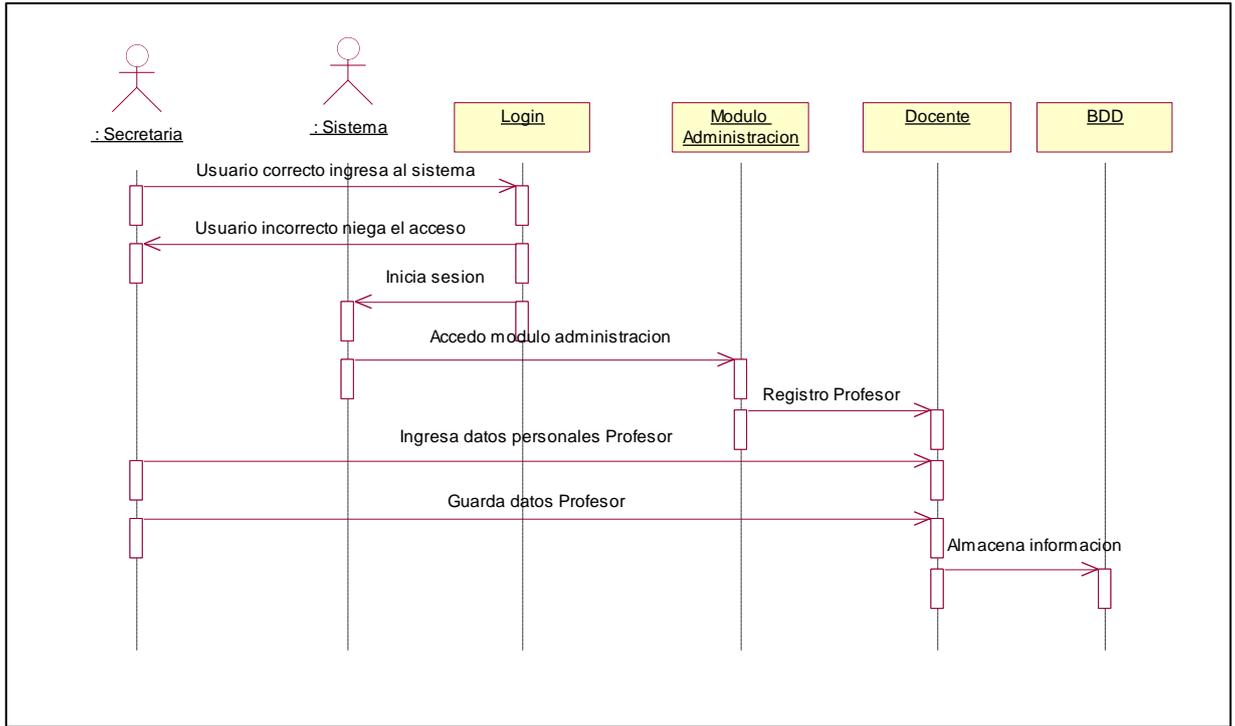
Ds_04 Registro de notas



Ds_05 Registro de Alumno



Ds_06 Registro de Profesor



Ds_07 Reporte

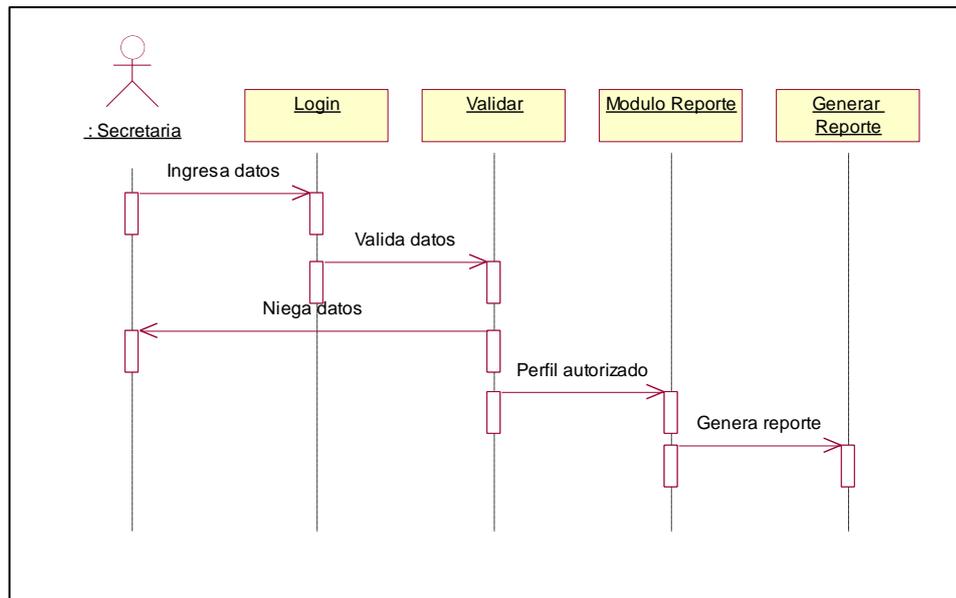
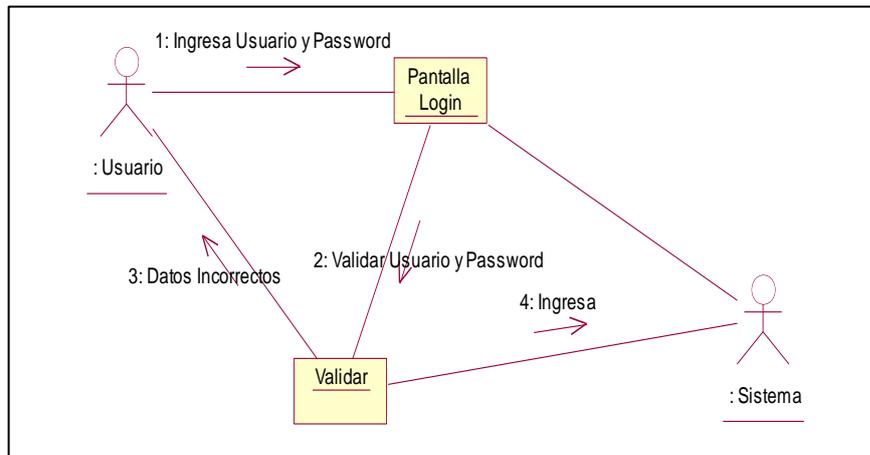


Diagrama de Colaboración

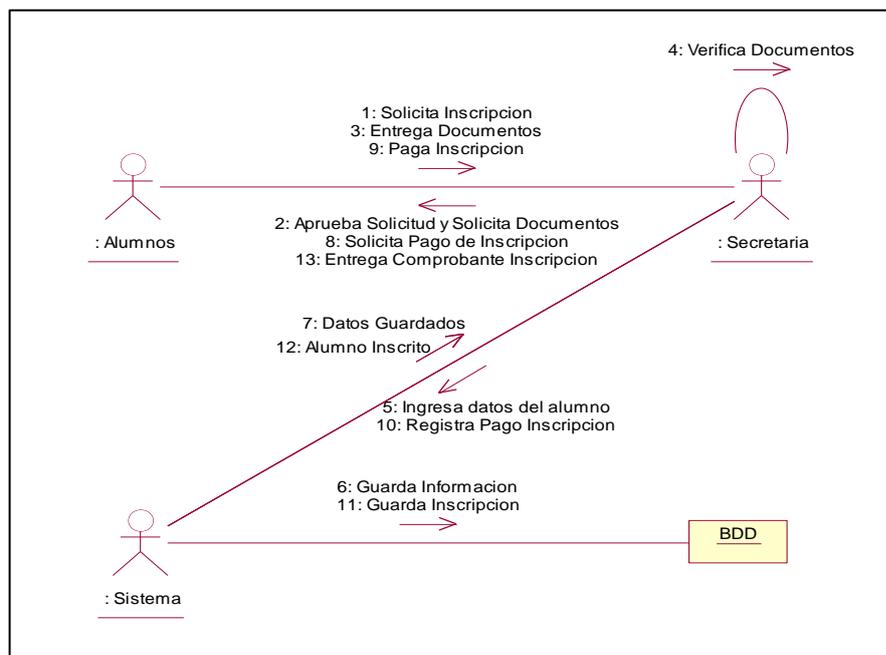
Al momento de realizar los diagramas de Secuencia los cuales son los pasos a seguir, en los diagramas de colaboración, se visualizara los pasos a seguir pero en mayor especificación que los diagramas de secuencia, en dichos diagramas se contempla el direccionamiento que debe realizar el actor a las diferentes actividades que lo realiza el sistema.



DC_01 Login

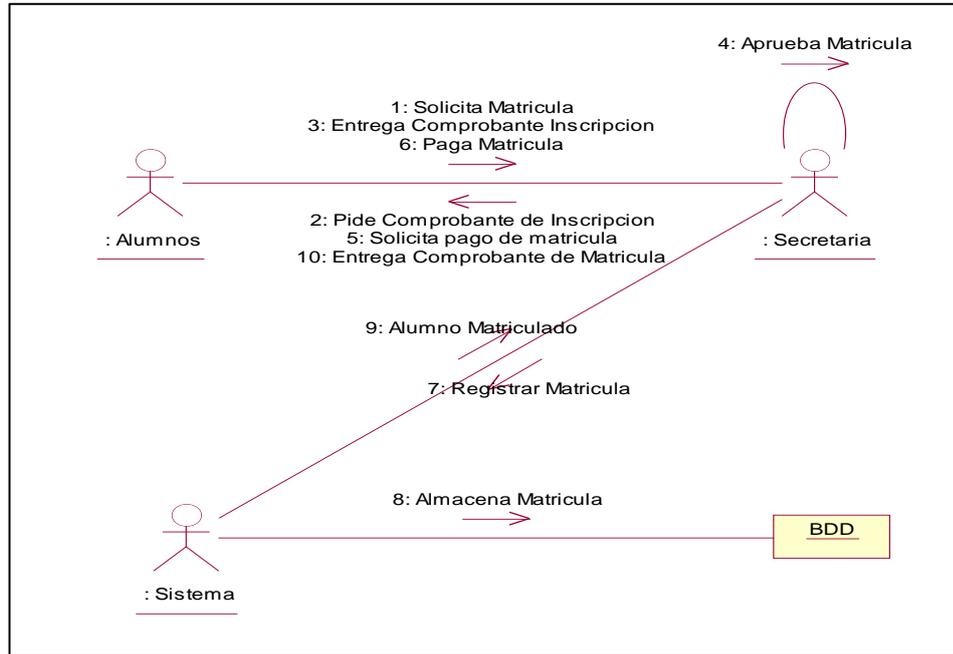


DC_02 Registro de Inscripción

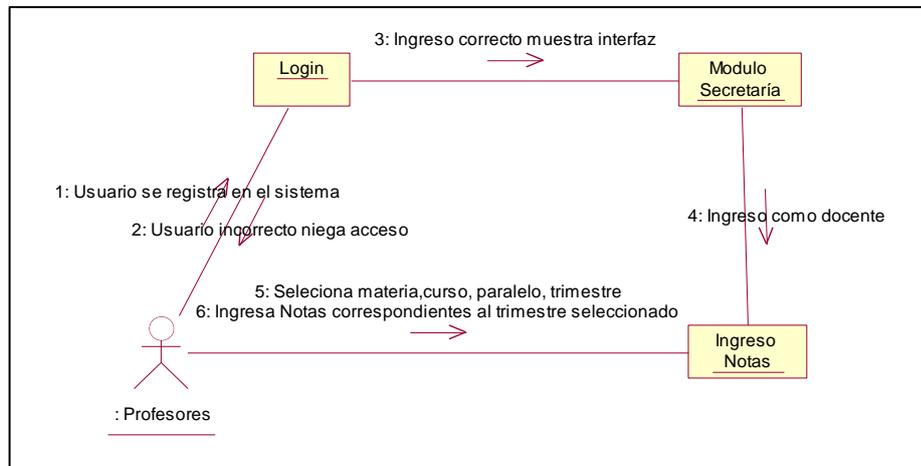




DC_03 Matriculación

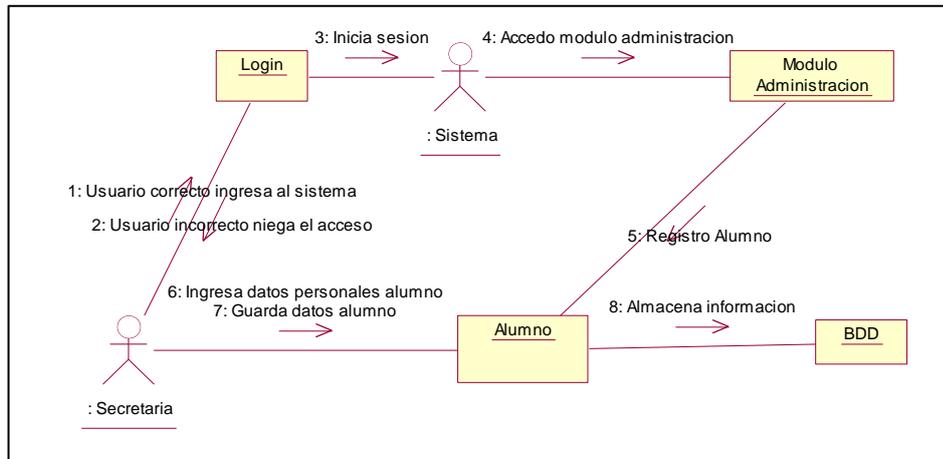


DC_04 Registro de Notas alumno

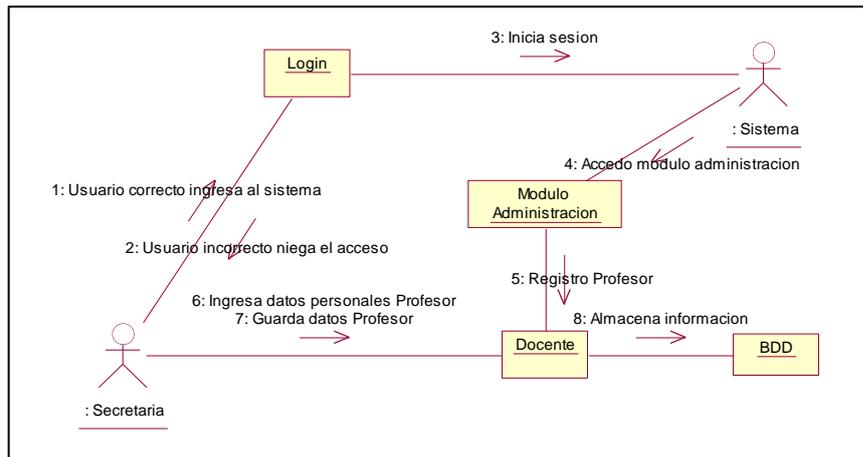




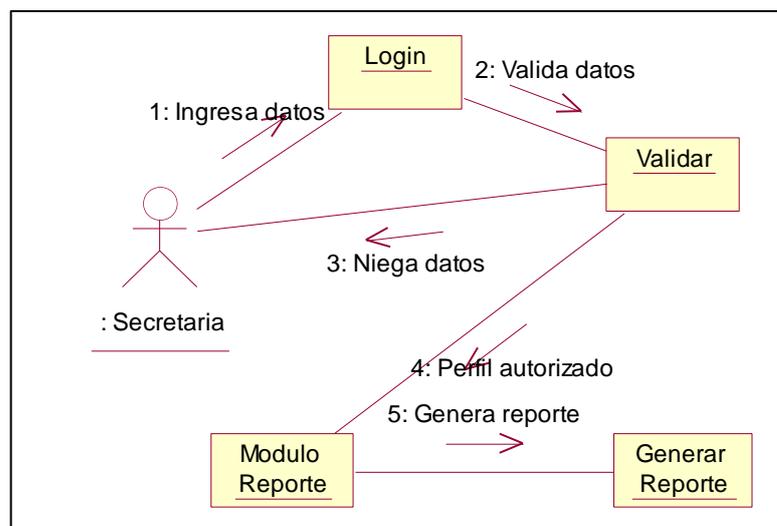
DC_05 Mantenimiento de Alumno



DC_06 Mantenimiento de Profesor



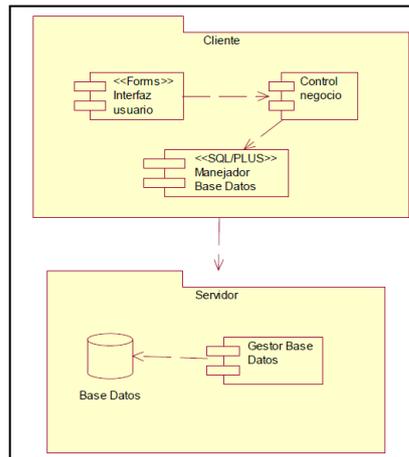
DC_07 Reporte



DIAGRAMAS DE COMPONENTES

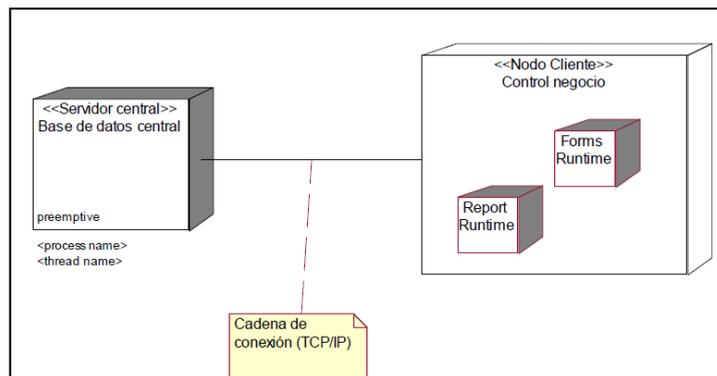
Muestra la organización y las dependencias entre un conjunto de componentes.

Cubren la vista de la implementación estática y se relacionan con los diagramas de clases ya que en un componente suele tener un o más clases, interfaces o colaboraciones.



DIAGRAMAS DE DESPLIEGUE

Representa la configuración de los nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos. Muestran la vista de despliegue estática de la arquitectura y se relacionan con los componentes ya que, los nodos contienen uno o más componentes.



Modelo Conceptual

El modelo conceptual requiere un análisis previo con el fin de conocer las entidades que van a intervenir en cada uno de los procesos. El modelo conceptual ayudará a plasmar en un papel el diseño que permitirá relacionar las diferentes entidades.



INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

Como entidades principales tenemos la de matrícula que es la que lleva la información del alumno, a los diferentes movimientos realizados llevados a cabo por los diferentes usuarios del sistema y generara reportes de los mismos.

Ver en la siguiente página.

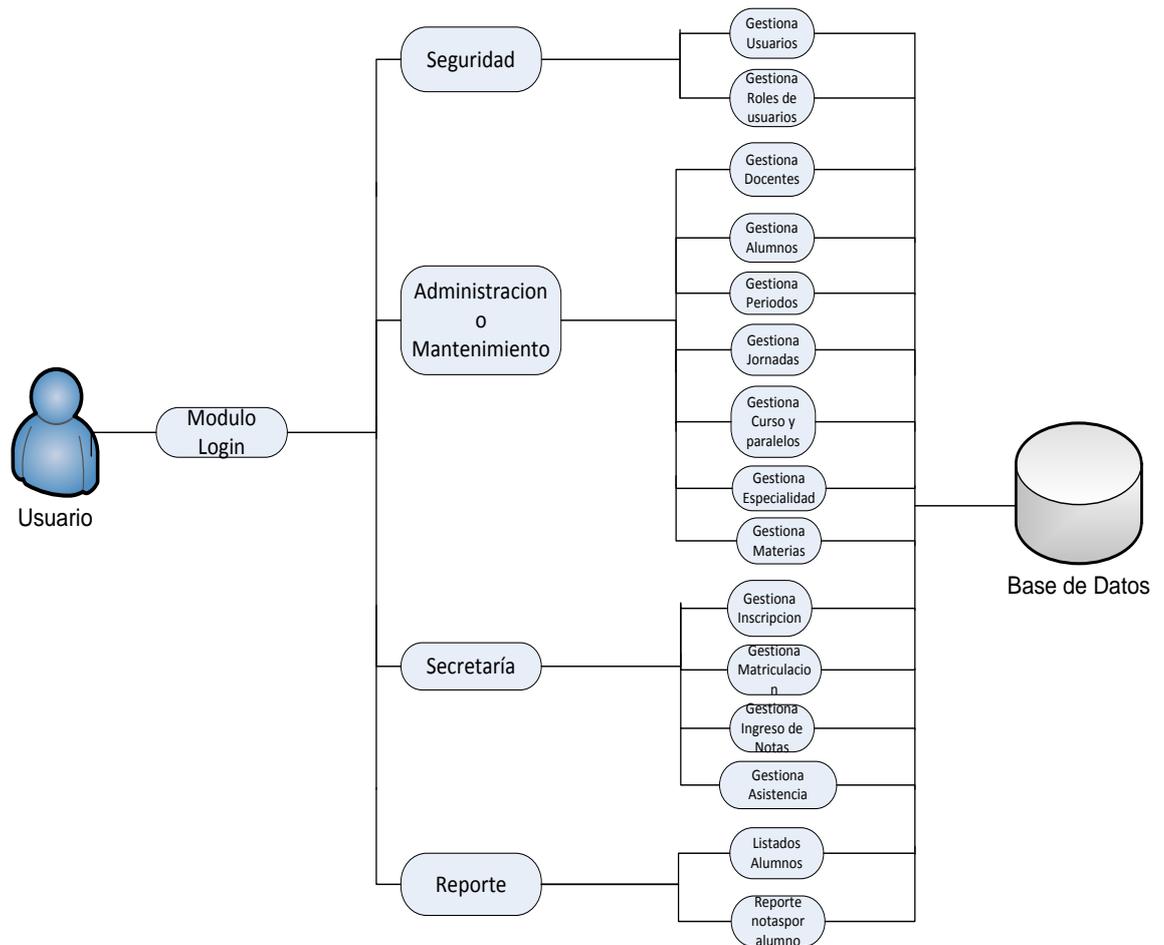


Diseño Detallado

Diseño Detallado de Módulos

A continuación se presenta el detalle del sistema definido en la arquitectura lógica y su correspondiente subdivisión en módulos, identificando las correlaciones entre cada uno.

SISTEMA DE CONTROL ACADÉMICO



Sistema de Control Académico

El Sistema de control académico es al que puede acceder un usuario registrado, y le permite, a través de formularios, insertar datos de docentes, alumnos, materias, especialidades, niveles o cursos y generar inscripciones y matriculas, ingreso de notas del alumno, controlar la asistencia, mismos que al ser enviados son procesados en las clases designadas para el manejo de datos, pasando por el proceso de validaciones y terminando con el almacenamientos en la base de datos.



Formulario de Administración

Parte que muestra un formulario, en el que el usuario puede ingresar datos personales de docentes y alumnos como: Cedula, Nombre, Dirección, entre otros. Datos de periodos, jornadas, materias, especialidad, curso como: Nombre, fecha, descripción, entre otros. Permite la modificación de todos de los datos ingresados.

1.1 Gestión de Docentes

Módulo que permite la creación así como la modificación de los docentes ya existentes en la base de datos.

1.2 Gestión de Alumnos

Módulo que permite la creación así como la modificación de los datos de los alumnos ya existentes en la base de datos.

1.3 Gestión de Periodos

Módulo que permite la creación así como la modificación de los datos de los periodos ya existentes en la base de datos.

1.4 Gestión de Jornadas

Módulo que permite la creación así como la modificación de los datos de las jornadas ya existentes en la base de datos.

1.5 Gestión de Especialidades

Módulo que permite la creación así como la modificación de los datos de las especialidades ya existentes en la base de datos.

1.6 Gestión de Cursos

Módulo que permite la creación así como la modificación de los datos de los cursos o niveles ya existentes en la base de datos.

1.7 Gestión de Materias

Módulo que permite la creación así como la modificación de los datos de las materias ya existentes en la base de datos.

Formulario de Secretaría

Permite al usuario registrado poder acceder a la modificación de los stocks previo proceso realizado en ingresos por adquisición o en devoluciones a proveedor.

1.8 Gestión de Inscripción

Módulo que permite la creación así como la modificación de la inscripción de los alumnos ya existentes en la base de datos.

1.9 Gestión de Matriculación

Módulo que permite la matriculación de los alumnos en la especialidad, curso, paralelo respectivo y así mismo su modificación los datos ya existentes en la base de datos.

1.10 Gestión de Ingreso de Notas



Módulo que permite el asentamiento de notas del alumnado de los distintos curso, paralelo y especialidades así como la modificación de las notas existentes en la base de datos.

1.11 Gestión de Asistencia

Modulo que permite registrar la asistencia de los alumnos a clase, así como también la justificación de una falta.

Formulario de Seguridad

Permite la creación de nuevos usuarios con su respectivo rol ya sea administrador o usuario.

1.12 Gestión de Usuarios

Módulo que permite la creación así como la modificación de usuarios.

1.13 Gestión de Roles

Módulo que permite la creación y asignación de nuevos roles para los usuarios si se lo requiere por lo general son administrador, profesor, secretaria, así como la modificación de dichos roles.

Formulario de Reportes

Modulo que permite al usuario del sistema generar listados de alumnos por curso, y reporte de notas del alumnado, reporte de asistencia así como la impresión de dichos reportes.

MANUAL DE USUARIO

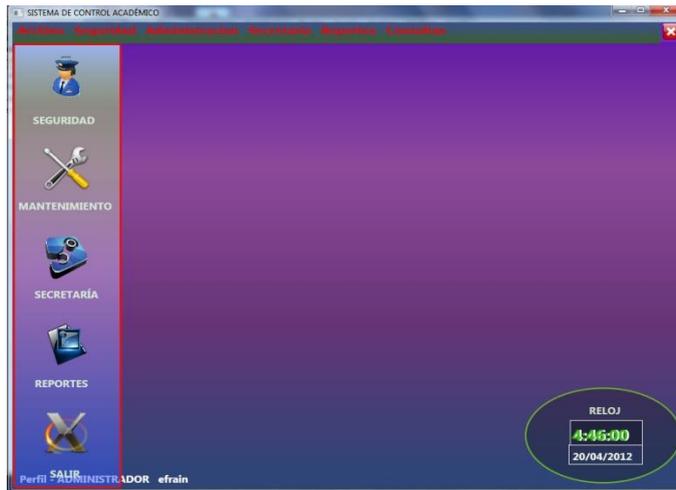
Formulario Inicial

Formulario de Login es el formulario inicial del sistema, el mismo que solicitara al usuario ingresar el nombre de usuario y la clave asignada para poder acceder al sistema.

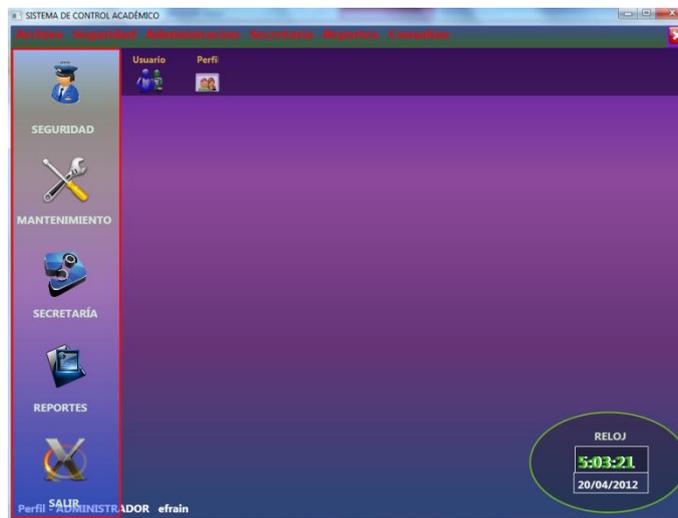




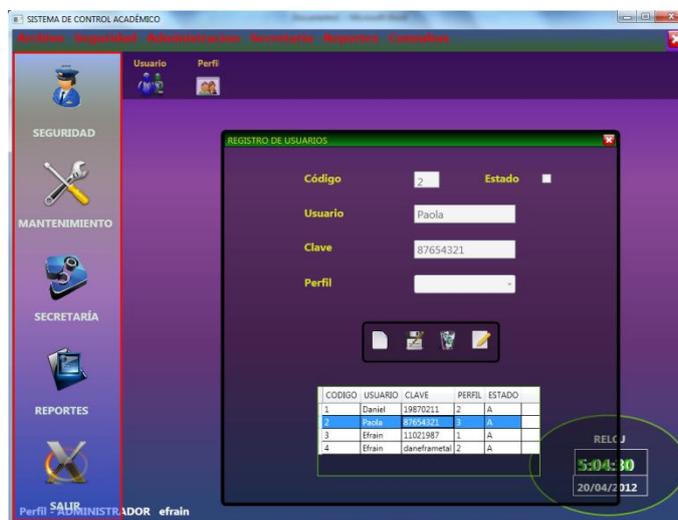
Formulario Principal



Pantalla principal del sistema con el Menú de Seguridad.



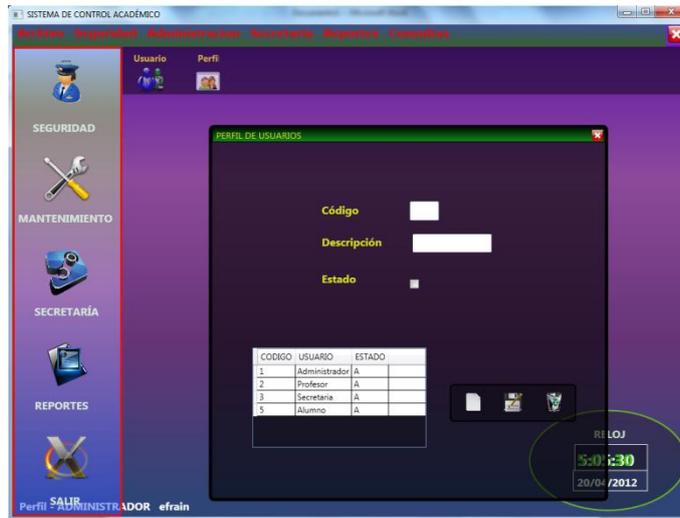
Pantalla principal del sistema con el Menú de Seguridad y formulario de registro de usuarios



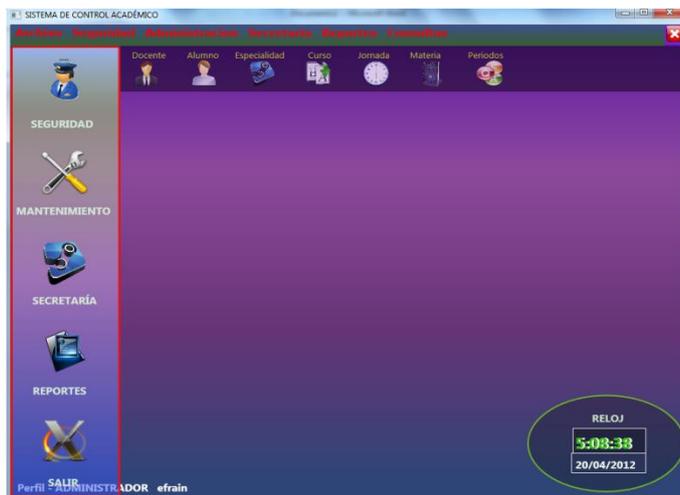


INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

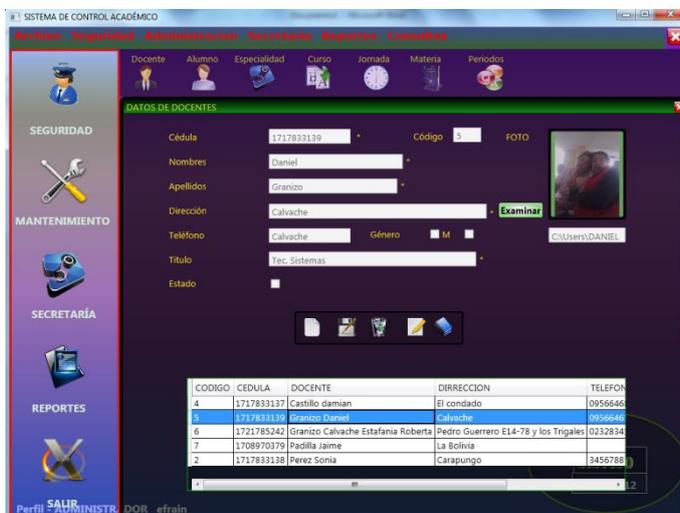
Pantalla principal del sistema con el Menú de Seguridad y formulario de perfil de usuarios



Pantalla principal del sistema con el Menú de Administración o Mantenimiento.



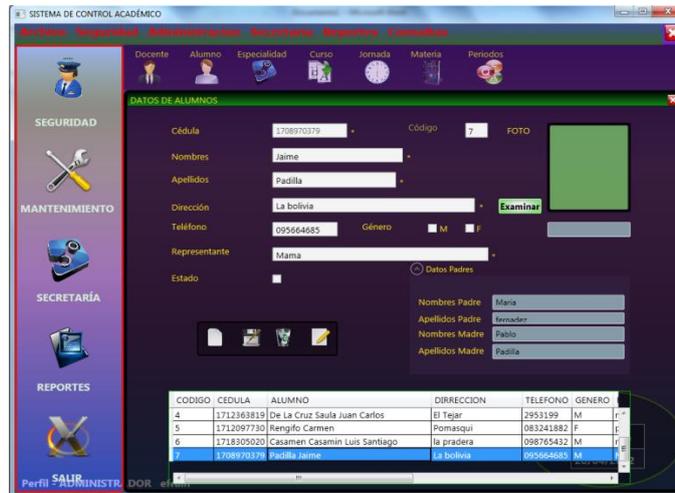
Pantalla principal del sistema con el formulario de registro de docentes.



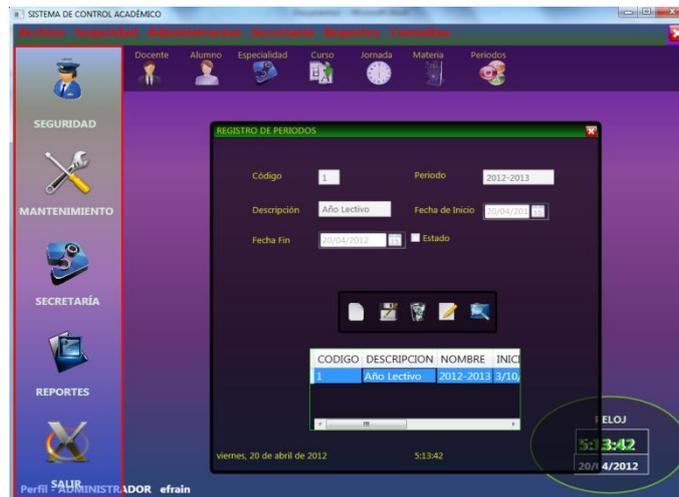


INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

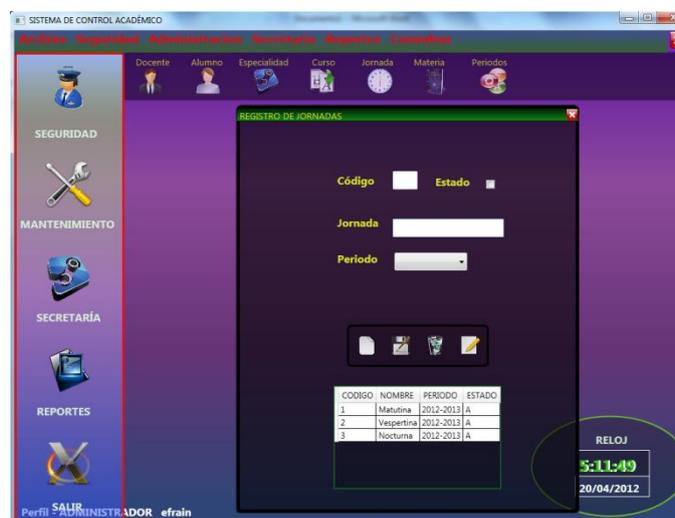
Pantalla principal del sistema con el formulario de registro de alumnos.



Pantalla principal del sistema con el formulario de registro de periodos.



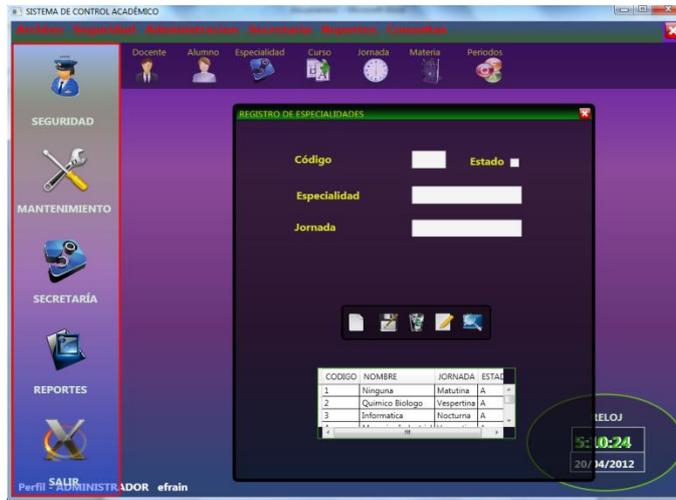
Pantalla principal del sistema con el formulario de registro de jornadas.



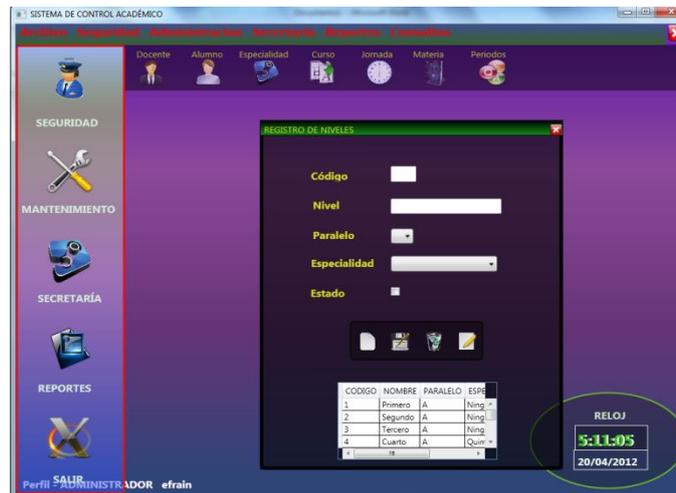


INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

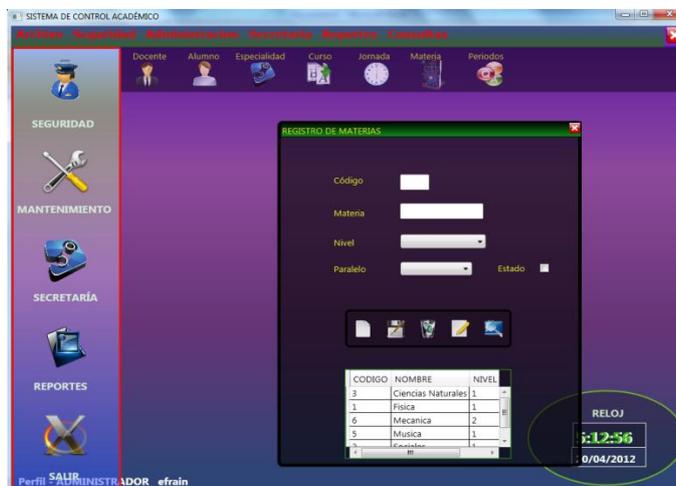
Pantalla principal del sistema con el formulario de registro de especialidades.



Pantalla principal del sistema con el formulario de registro de niveles.



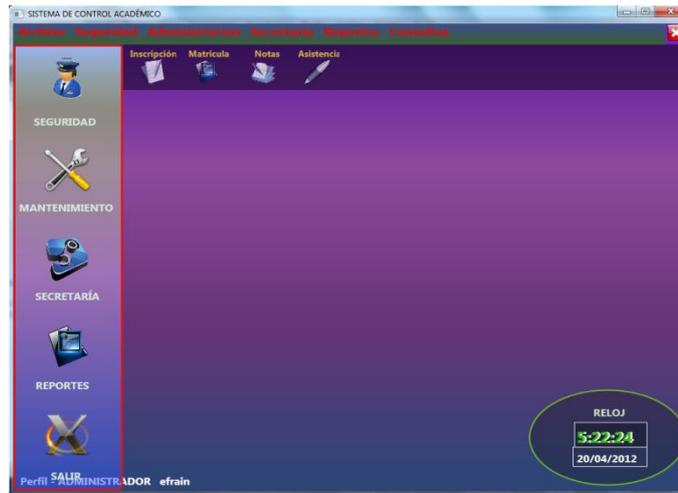
Pantalla principal del sistema con el formulario de registro de materias.



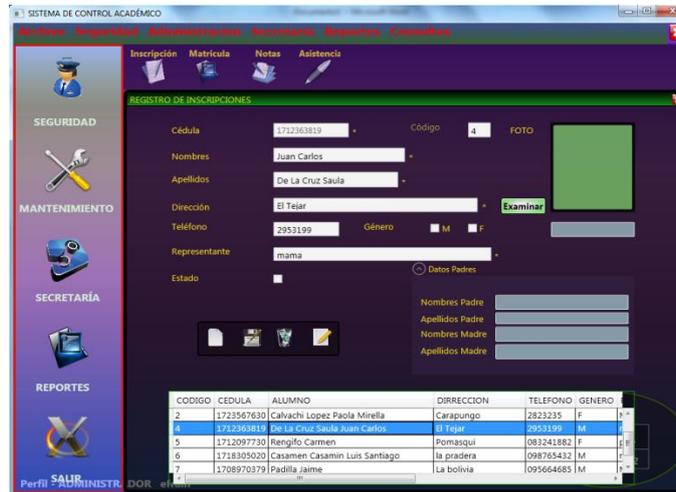


INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

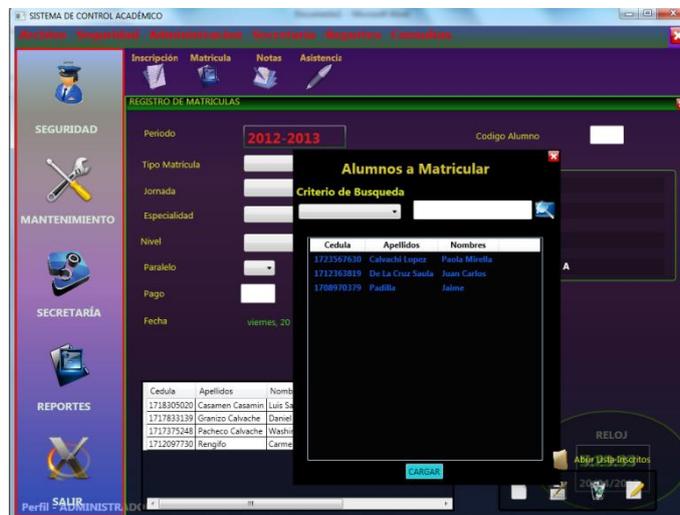
Pantalla principal del sistema con el Menú de secretaria



Pantalla principal del sistema con el formulario de registro de inscripciones.



Pantalla principal del sistema con el formulario de alumnos a matricular.



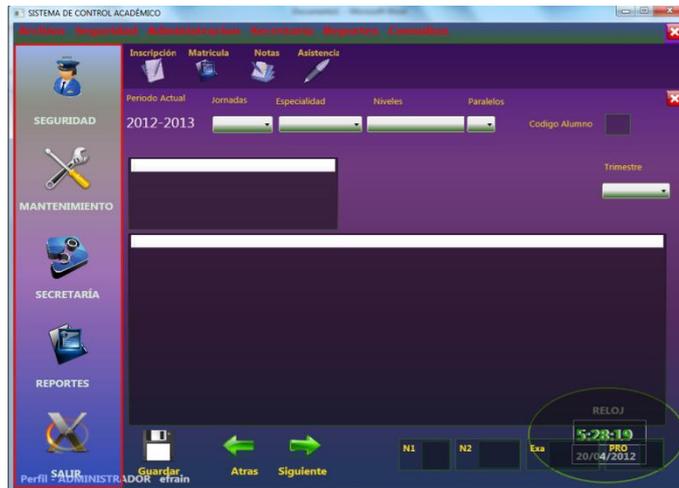


INSTITUTO TECNOLÓGICO SUPERIOR CORILLERA

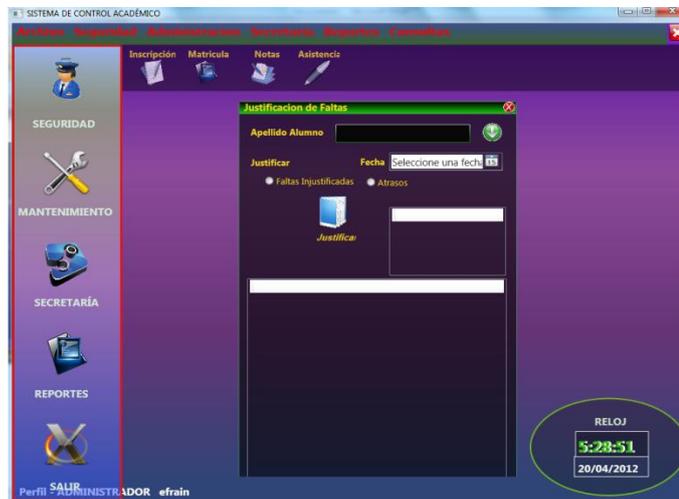
Pantalla principal del sistema con el formulario matricula después de haber cargado el alumno.



Pantalla principal del sistema con el formulario de ingreso de notas

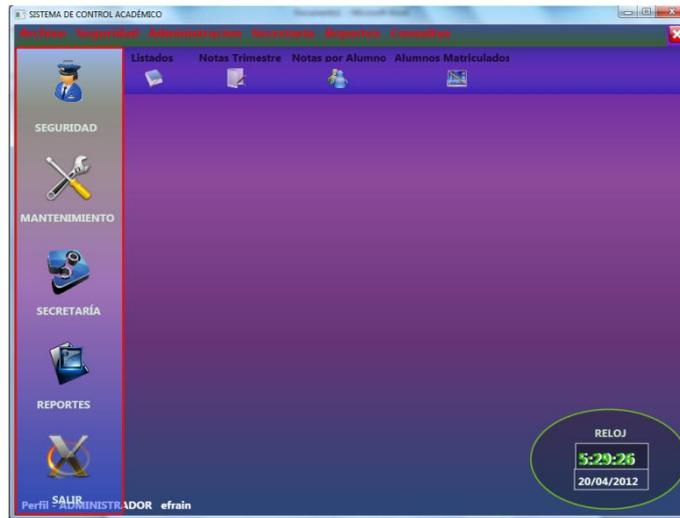


Pantalla principal del sistema con el formulario de justificación de faltas

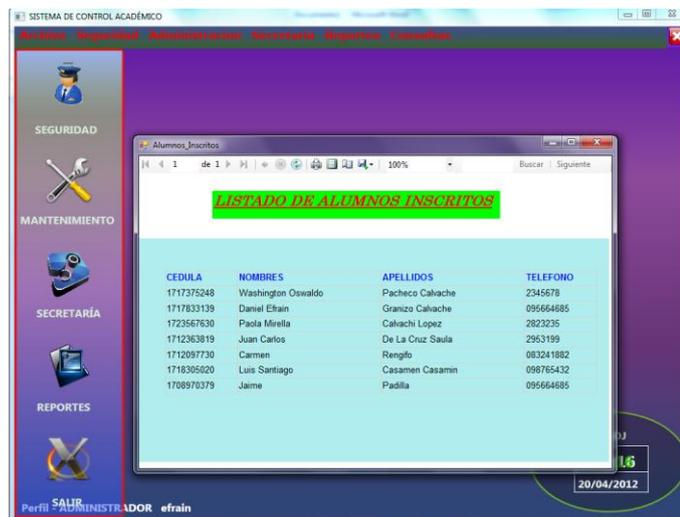




Pantalla principal del sistema con el formulario de reporte.



Pantalla principal del sistema con un reporte.

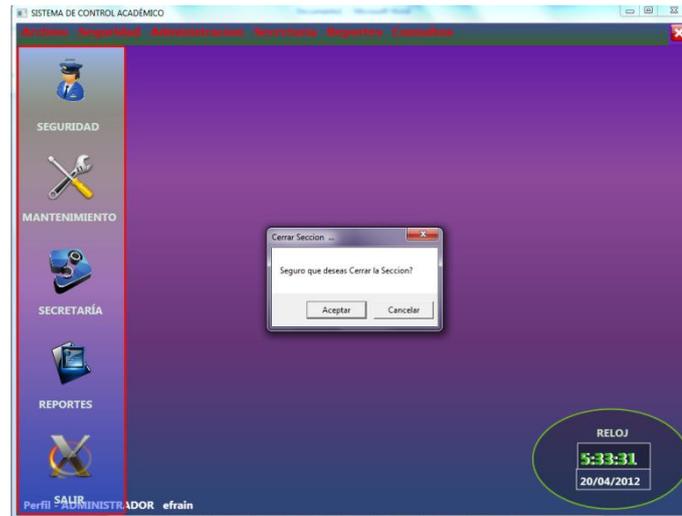


Pantalla principal del sistema con el formulario de búsqueda de alumno.

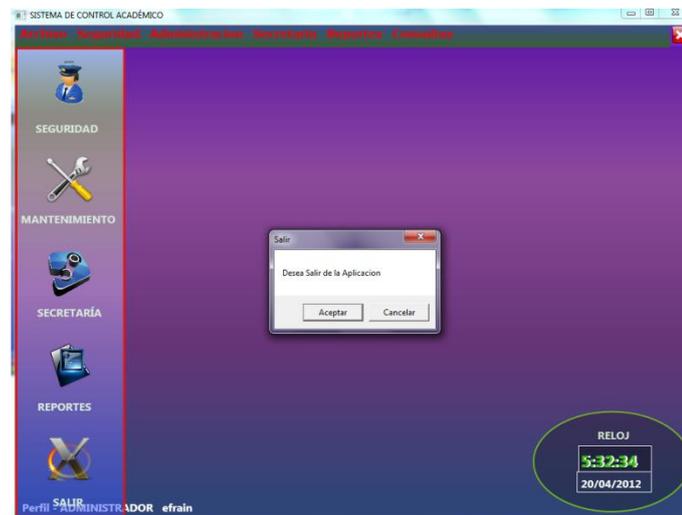




Pantalla principal del sistema con mensaje de cerrar sesión.



Pantalla principal del sistema con mensaje de salir de la aplicación.





Anexo N°13

Manual Técnico



MANUAL TÉCNICO

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;
using System.Data.Common;
using System.Data.Linq;
using System.Windows.Forms;
using Microsoft.Win32;
using System.IO;
using System.Drawing;
using Acceso_Datos;
using Logica_Negocio;

namespace SISACA
{
    public partial class MainWindow : Window
    {
        Validacion validar = new Validacion();
        public MainWindow()
        {
            InitializeComponent();
            Cerrar_Grid();
        }

        #region Menu
        private void M_Administrador_Click(object sender, System.Windows.RoutedEventArgs e)
        {
            Cerrar_Grid();
            Grd_Menu_Mantenimiento.Visibility = Visibility.Collapsed;
            Grd_Menu_Seguridad.Visibility = Visibility.Visible;
        }

        private void M_Registro_Click(object sender, System.Windows.RoutedEventArgs e)
        {
            Cerrar_Grid();
            Grd_Menu_Seguridad.Visibility = Visibility.Collapsed;
            Grd_Menu_Administrativo.Visibility = Visibility.Collapsed;
            Grd_Menu_Mantenimiento.Visibility = Visibility.Visible;
        }

        private void M_Administrativo_Click(object sender, System.Windows.RoutedEventArgs e)
        {
            Cerrar_Grid();
            Grd_Menu_Seguridad.Visibility = Visibility.Collapsed;
            Grd_Menu_Mantenimiento.Visibility = Visibility.Collapsed;
            Grd_Menu_Administrativo.Visibility = Visibility.Visible;
        }

        private void Img_Mantenimiento_MouseLeftButtonDown(object sender,
            System.Windows.Input.MouseButtonEventArgs e)
```



```
{
    Cerrar_Grid();
    Grd_Menu_Seguridad.Visibility = Visibility.Collapsed;
    Grd_Menu_Administrativo.Visibility = Visibility.Collapsed;
    Grd_Menu_Mantenimiento.Visibility = Visibility.Visible;
}
private void Img_Seguridad_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Grd_Menu_Administrativo.Visibility = Visibility.Collapsed;
    Grd_Menu_Mantenimiento.Visibility = Visibility.Collapsed;
    Grd_Menu_Seguridad.Visibility = Visibility.Visible;
}
private void Img_Administrativo_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Grd_Menu_Seguridad.Visibility = Visibility.Collapsed;
    Grd_Menu_Mantenimiento.Visibility = Visibility.Collapsed;
    Grd_Menu_Administrativo.Visibility = Visibility.Visible;
}
private void Img_Cerrar_Aplicacion_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    if (System.Windows.Forms.MessageBox.Show("Desea Salir de la Aplicacion",
("Salir"), MessageBoxButton.OKCancel) == System.Windows.Forms.DialogResult.OK)
    {
        this.Close();
    }
}
private void Img_Salir_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    if (System.Windows.Forms.MessageBox.Show("Desea Salir de la Aplicacion",
("Salir"), MessageBoxButton.OKCancel) == System.Windows.Forms.DialogResult.OK)
    {
        this.Close();
    }
}
}
#endregion

#region Sub_Menu
private void Cerrar_Grid()
{
    Grd_Periodos.Visibility = Visibility.Collapsed;
    Grd_Jornadas.Visibility = Visibility.Collapsed;
    Grd_Especialidad.Visibility = Visibility.Collapsed;
    Grd_Niveles.Visibility = Visibility.Collapsed;
    Grd_Materias.Visibility = Visibility.Collapsed;
    Grd_Matriculas.Visibility = Visibility.Collapsed;
    Grd_Parametros.Visibility = Visibility.Collapsed;
    Grd_Profesores.Visibility = Visibility.Collapsed;
    Grd_Alumnos.Visibility = Visibility.Collapsed;
    Grd_Inscripciones.Visibility = Visibility.Collapsed;
    Brd_Niveles.Visibility = Visibility.Collapsed;
    Brd_Registro_Alumno.Visibility = Visibility.Collapsed;
    Brd_Profesores.Visibility = Visibility.Collapsed;
    Brd_Matrias.Visibility = Visibility.Collapsed;
}
```



```
        Brd_Registro_Profesor.Visibility = Visibility.Collapsed;
        Brd_Periodos.Visibility = Visibility.Collapsed;
        Brd_Perfil.Visibility = Visibility.Collapsed;
        Brd_Usuarios.Visibility = Visibility.Collapsed;
    }
    private void Img_Periodo_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
    {
        Cerrar_Grid();
        Brd_Periodos.Visibility=Visibility.Visible;
        Lbl_Tit_Periodo.Content = "REGISTRO DE PERIODOS";
        Dtp_Per_F_Inicio.SelectedDate = DateTime.Now.ToLocalTime();
        Dtp_Per_F_Fin.SelectedDate = DateTime.Now.ToLocalTime();
        Lbl_Fecha.Content = DateTime.Now.ToLongDateString();
        Grd_Periodos.Visibility = Visibility.Visible;
        Bloquear_Caja_Texto();
        Cargar_Periodos();
    }
    private void Img_Jornada_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
    {
        Cerrar_Grid();
        Cargar_Jornadas();
        Lbl_Tit_Jornadas.Content = "REGISTRO DE JORNADAS";
        Grd_Jornadas.Visibility = Visibility.Visible;
    }
    private void Img_Especialidad_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
    {
        Cerrar_Grid();
        Lbl_Tit_Especialidad.Content = "REGISTRO DE ESPECIALIDADES";
        Grd_Especialidad.Visibility = Visibility.Visible;
        BloquearCajasTextoEspecialidad();
        Cargar_Especialidad();
    }
    private void Img_curso_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
    {
        Cerrar_Grid();
        Brd_Niveles.Visibility = Visibility.Visible;
        Cargar_Niveles();
        Lbl_Tit_Niveles.Content = "REGISTRO DE NIVELES";
        Grd_Niveles.Visibility = Visibility.Visible;
    }
    private void Img_Materia_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
    {
        Cerrar_Grid();
        Lbl_Tit_Materias.Content = "REGISTRO DE MATERIAS";
        Brd_Materias.Visibility = Visibility.Visible;
        Grd_Materias.Visibility = Visibility.Visible;
        Cargar_Materias();
    }
    private void Img_Matricula_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
    {
        Cerrar_Grid();
        Lbl_Tit_Matriculas.Content = "REGISTRO DE MATRICULAS";
        Lbl_FechaMatriculacion.Content = DateTime.Now;
        Grd_Matriculas.Visibility = Visibility.Visible;
    }
}
```



```
Cargar_Matriculacion();
Cargar_Datos_Matricula();
}

private void Img_Inscripcion_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Cargar_Alumno();
    Lbl_Tit_Reg_Alumno.Content = "REGISTRO DE INSCRIPCIONES";
    Brd_Registro_Alumno.Visibility = Visibility.Visible;
    Btn_Nuevo_Alumno.IsEnabled = false;
}
private void Img_Alumnos_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Cargar_Busqueda_Alumno();
    Lbl_Tit_Alumno.Content = "BUSQUEDA DE ALUMNOS";
    Grd_Alumnos.Visibility = Visibility.Visible;
}

private void Img_Docentes_MouseLeftButtonUp(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Cargar_Busqueda_Profesor();
    Brd_Profesores.Visibility = Visibility.Visible;
    Grd_Profesores.Visibility = Visibility.Visible;
}
private void Img_Usuarios_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Cargar_Usuarios();
    Bloquear_Usuario();
    Lbl_Tit_Usuario.Content = "REGISTRO DE USUARIOS";
    Brd_Usuarios.Visibility = Visibility.Visible;
}
private void Img_Perfil_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Cargar_Perfil_Usurio();
    Lbl_Tit_Perfil.Content = "PERFIL DE USUARIOS";
    Brd_Perfil.Visibility = Visibility.Visible;
}
private void Img_IngresoNotas_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Grd_NotasDocentes.Visibility = Visibility.Visible;
}

#region Alumno
private void Cargar_Busqueda_Alumno()
{
    Alumnos_Logica Log_Alumnos = new Alumnos_Logica();
    Lst_Alumno.ItemsSource = Log_Alumnos.Obtener_Alumno();
    Btn_Buscar_Alumno.Tag = "";
}
```



```
Dgr_Alumno.ItemsSource = Log_Alumnos.Obtener_Alumno().Select(alu => new
{
    CODIGO = alu.Alu_Codigo,
    CEDULA = alu.Alu_Cedula,
    NOMBRES = alu.Alu_Nombres,
    APELLIDOS = alu.Alu_Apellidos,
    DIRRECCION = alu.Alu_Direccion,
    TELEFONO = alu.Alu_Telefono,
    GENERO = alu.Alu_Genero,
    REPRESENTANTE = alu.Alu_Representante,
    ESTADO = alu.Alu_Estado
}).ToList();
Btn_Buscar_Alumno.Tag = "";
}
private void Cargar_Alumno()
{
    Alumnos_Logica Alumnos = new Alumnos_Logica();
    Lst_RAlumno.ItemsSource = Alumnos.Obtener_Alumno();
    Btn_Nuevo_Alumno.Tag = "";
    Dgr_RAlumno.ItemsSource = Alumnos.Obtener_Alumno().Select(alu => new
    {
        CODIGO = alu.Alu_Codigo,
        CEDULA = alu.Alu_Cedula,
        NOMBRES = alu.Alu_Nombres,
        APELLIDOS = alu.Alu_Apellidos,
        DIRRECCION = alu.Alu_Direccion,
        TELEFONO = alu.Alu_Telefono,
        GENERO = alu.Alu_Genero,
        REPRESENTANTE = alu.Alu_Representante,
        FOTO = Img_Alu_Foto,
        ESTADO = alu.Alu_Estado
    }).ToList();
    Btn_Nuevo_Alumno.Tag = "";
}
private void Buscar_AlumnoXCedula(string Cedula)
{
    Alumnos_Logica Log_Alumno = new Alumnos_Logica();
    Lst_Alumno.ItemsSource = Log_Alumno.Buscar_AlumnoXCedula(Cedula);
    Dgr_Alumno.ItemsSource = Log_Alumno.Buscar_AlumnoXCedula(Cedula).Select(alu => new
    {
        CEDULA = alu.Alu_Cedula,
        NOMBRES = alu.Alu_Nombres,
        APELLIDOS = alu.Alu_Apellidos,
        DIRRECCION = alu.Alu_Direccion,
        TELEFONO = alu.Alu_Telefono,
        ESTADO = alu.Alu_Estado
    }).ToList();
    Btn_Buscar_Alumno.Tag = "";
}
private void Buscar_AlumnoXAPELLIDO(string Apellido)
{
    Alumnos_Logica BuscaApellido = new Alumnos_Logica();
    Lst_Alumno.ItemsSource = BuscaApellido.Buscar_AlumnoXAPELLIDO(APELLIDO);
    Dgr_Alumno.ItemsSource =
    BuscaApellido.Buscar_AlumnoXAPELLIDO(APELLIDO).Select(alu => new
    {
        CEDULA = alu.Alu_Cedula,
        NOMBRES = alu.Alu_Nombres,
        APELLIDOS = alu.Alu_Apellidos,
        DIRRECCION = alu.Alu_Direccion,
```



```
        TELEFONO = alu.Alu_Telefono,
        ESTADO = alu.Alu_Estado
    }).ToList();
    Btn_Buscar_Alumno.Tag = "";
}
private void Bloquear_Alumno()
{
    Txt_RAlu_Codigo.Enabled = false;
    Txt_RAlu_Cedula.Enabled = false;
    Txt_RAlu_Nombres.Enabled = false;
    Txt_RAlu_Apellidos.Enabled = false;
    Txt_RAlu_Direccion.Enabled = false;
    Txt_RAlu_Representante.Enabled = false;
    Txt_RAlu_Telefono.Enabled = false;
    Chk_RAlu_Estado.Enabled = false;
    Chk_RAlu_Genero_F.Enabled = false;
    Chk_RAlu_Genero_M.Enabled = false;
}
private void Desbloquear_Alumno()
{
    Txt_RAlu_Cedula.Enabled = true;
    Txt_RAlu_Nombres.Enabled = true;
    Txt_RAlu_Apellidos.Enabled = true;
    Txt_RAlu_Direccion.Enabled = true;
    Txt_RAlu_Representante.Enabled = true;
    Txt_RAlu_Telefono.Enabled = true;
    Chk_RAlu_Estado.Enabled = true;
    Chk_RAlu_Genero_F.Enabled = true;
    Chk_RAlu_Genero_M.Enabled = true;
}
private void Txt_Buscar_Alumno_TextChanged(object sender,
System.Windows.Controls.TextChangedEventArgs e)
{
    if (Txt_Buscar_Alumno.Text != "")
    {
        try
        {
            string op = Cbo_OpBuscar_Alumno.SelectedItem.ToString();
            switch (op)
            {
                case "Cedula":
                    Buscar_AlumnoXCedula(Txt_Buscar_Alumno.Text);
                    break;
                case "Apellido":
                    Buscar_AlumnoXApellido(Txt_Buscar_Alumno.Text);
                    break;
            }
        }
        catch (Exception)
        {
            System.Windows.Forms.MessageBox.Show("Ingrese opcion de Búsqueda");
        }
    }
    else
    {
        Cargar_Busqueda_Alumno();
    }
}
```



```
private int SeqAlumno_Id()
{
    int codigo = 0;
    AcademicoDataContext dc = new AcademicoDataContext();
    try
    {
        codigo = dc.Tbl_Alumno.Max(cod => cod.Alu_Codigo + 1);
    }
    catch (Exception)
    {
        codigo = 1;
    }
    return codigo;
}
private void Lst_Alumno_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Grd_Alumnos.DataContext = Lst_Alumno.SelectedItem;
}
private void Dgr_Alumno_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Lst_Alumno.SelectedIndex = Dgr_Alumno.SelectedIndex;
}
private void Lst_RAlumno_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Grd_Registro_Alumno.DataContext = Lst_RAlumno.SelectedItem;
}
private void Dgr_RAlumno_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Lst_RAlumno.SelectedIndex = Dgr_RAlumno.SelectedIndex;
}
private void Btn_Registra_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cerrar_Grid();
    Lbl_Tit_Reg_Alumno.Content = "INSCRIPCIÓN DE ALUMNOS";
    Brd_Registro_Alumno.Visibility = Visibility.Visible;
    Cargar_Alumno();
    Bloquear_Alumno();
    Txt_RAlu_Codigo.Clear();
    Txt_RAlu_Cedula.Clear();
    Txt_RAlu_Nombres.Clear();
    Txt_RAlu_Apellidos.Clear();
    Txt_RAlu_Direccion.Clear();
    Txt_RAlu_Telefono.Clear();
    Txt_RAlu_Representante.Clear();
    Chk_RAlu_Genero_F.IsChecked = false;
    Chk_RAlu_Genero_M.IsChecked = false;
    Chk_RAlu_Estado.IsChecked = false;
}
private void NuevoAlumno()
{
    Desbloquear_Alumno();
    Txt_RAlu_Codigo.Text = SeqAlumno_Id().ToString();
    Txt_RAlu_Cedula.Clear();
    Txt_RAlu_Nombres.Clear();
    Txt_RAlu_Apellidos.Clear();
}
```



```
Txt_RAlu_Direccion.Clear();
Txt_RAlu_Telefono.Clear();
Txt_RAlu_Representante.Clear();
Txt_Alu_Path_Foto.Clear();
Chk_RAlu_Genero_F.IsChecked = false;
Chk_RAlu_Genero_M.IsChecked = false;
Chk_RAlu_Estado.IsChecked = false;
Brd_Alu_Foto.Visibility = Visibility.Collapsed;
Btn_Nuevo_Alumno.Tag = "Nuevo";
}
private void Btn_Nuevo_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    NuevoAlumno();
}
private void Btn_Guardar_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    if (Btn_Nuevo_Alumno.Tag.ToString() == "Nuevo")
    {
        Tbl_Alumno NuevoAlumno = new Tbl_Alumno();
        NuevoAlumno.Alu_Codigo =
Convert.ToInt32(Txt_RAlu_Codigo.Text);
        NuevoAlumno.Alu_Cedula = Txt_RAlu_Cedula.Text;
        NuevoAlumno.Alu_Nombres = Txt_RAlu_Nombres.Text;
        NuevoAlumno.Alu_Apellidos = Txt_RAlu_Apellidos.Text;
        NuevoAlumno.Alu_Direccion = Txt_RAlu_Direccion.Text;
        NuevoAlumno.Alu_Telefono = Txt_RAlu_Telefono.Text;
        NuevoAlumno.Alu_Foto = Txt_Alu_Path_Foto.Text;
        NuevoAlumno.Alu_Representante = Txt_RAlu_Representante.Text;
        if (Chk_RAlu_Genero_M.IsChecked == true)
        {
            NuevoAlumno.Alu_Genero = 'M';
        }
        else
        {
            if (Chk_RAlu_Genero_F.IsChecked == true)
            {
                NuevoAlumno.Alu_Genero = 'F';
            }
        }
        if (Chk_RAlu_Estado.IsChecked == true)
        {
            NuevoAlumno.Alu_Estado = 'A';
        }
        else
        {
            NuevoAlumno.Alu_Estado = 'I';
        }
        if (Lst_RAlumno.Items.Count > 0)
        {
            Lst_RAlumno.SelectedIndex = 0;
        }
        Alumnos_Logica.dc.Refresh(RefreshMode.OverwriteCurrentValues,
Lst_RAlumno.SelectedItem);
    }
    Alumnos_Logica.dc.Tbl_Alumno.InsertOnSubmit(NuevoAlumno);
    try
    {
        Alumnos_Logica Log_Alumnos = new Alumnos_Logica();
```



```
        Log_Alumnos.Guardar_Alumno();
        Cargar_Alumno();
        System.Windows.Forms.MessageBox.Show("Registro Almacenado con Exito");
    }
    catch (Exception ex)
    {
        System.Windows.Forms.MessageBox.Show("Registro NO Almacenado" + ex.ToString());
    }
    Bloquear_Alumno();
}

else
{
    try
    {
        Alumnos_Logica Log_Alumnos = new Alumnos_Logica();
        Log_Alumnos.Guardar_Alumno();
        Btn_Nuevo_Alumno.Tag = "";
        Cargar_Alumno();
        System.Windows.Forms.MessageBox.Show("Registro Modificado Correctamente", Alumnos);
    }
    catch (Exception)
    {
        System.Windows.Forms.MessageBox.Show("Registro No Modificado Correctamente",
        "Alumnos");
    }
    Bloquear_Alumno();
}
}

private void Btn_Eliminar_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Alumnos_Logica dc.Tbl_Alumno.DeleteOnSubmit(((Tbl_Alumno)Lst_RAlumno.SelectedItem);
    try
    {
        if (Lst_RAlumno.SelectedItem==null)
        {
            System.Windows.Forms.MessageBox.Show("Seleccione un Registro de la Lista");
            return;
        }
        if (System.Windows.Forms.MessageBox.Show("Eliminar Registro de Alumno\n¿Esta seguro
de eliminar registro de alumno
ahora?"), "Alumno", MessageBoxButtons.OKCancel)==System.Windows.Forms.DialogResult.OK)
        {
            AcademicoDataContext dc = new AcademicoDataContext();
            Tbl_Alumno EliminarAlumno = new Tbl_Alumno();
            dc.ExecuteCommand("Update Tbl_Alumno set Alu_Estado={0} where Alu_Codigo={1}",
            new object[]
            {
                EliminarAlumno.Alu_Estado='I',
                EliminarAlumno.Alu_Codigo=(((Tbl_Alumno)Lst_RAlumno.SelectedItem).Alu_Codigo)
            });
            Cargar_Alumno();
            System.Windows.Forms.MessageBox.Show("Registro Alumnos Eliminado");
        }
    }
    catch (Exception)
    {
        System.Windows.Forms.MessageBox.Show("Registro Alumno No Eliminado");
    }
}
}
```



```
private void Btn_Modificar_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Desbloquear_Alumno();
}
private void Txt_RAlu_Cedula_LostFocus(object sender,
System.Windows.RoutedEventArgs e)
{
    if (Utilidades.Verificaciones.VerificaCedula(Txt_RAlu_Cedula.Text))
    { }
    else
    {
        System.Windows.Forms.MessageBox.Show("CEDULA INCORRECTA");
    }
}
private void Txt_RAlu_Cedula_KeyUp(object sender,
System.Windows.Input.KeyEventArgs e)
{
    validar.ValidoNumero(Convert.ToInt32(e.Key).ToString());
}
private void Txt_RAlu_Cedula_KeyDown(object sender,
System.Windows.Input.KeyEventArgs e)
{
    if (Convert.ToInt32(e.Key) == 6)
    {
        Txt_RAlu_Nombres.Focus();
    }
}
private void Cargar_DatosXCedula()
{
    if (Txt_Buscar_Alumno.Text.Length == 10 && Txt_Buscar_Alumno.Text != "")
    {
        if (System.Windows.Forms.MessageBox.Show("¿Registro no existe\n¿Desea registra
ahora?", "Alumno", MessageBoxButtons.OK) == System.Windows.Forms.DialogResult.OK)
        {
            Grd_Alumnos.Visibility = Visibility.Collapsed;
            Brd_Registro_Alumno.Visibility = Visibility.Visible;
            NuevoAlumno();
            Txt_RAlu_Cedula.Text = Txt_Buscar_Alumno.Text;
        }
    }
}
private void Btn_Buscar_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Cargar_DatosXCedula();
}
private void Btn_Examinar_Foto_Alumno_Click(object sender,
System.Windows.RoutedEventArgs e)
{
    BitmapDecoder bitdecoder;
    System.Windows.Forms.OpenFileDialog dialog1 = new
System.Windows.Forms.OpenFileDialog();
    dialog1.Filter = "JPG (*.jpg)|*.jpg|BMP (*.bmp)|*.bmp|PNG
(*.png)|*.png|Todos los Archivos (*.*)|*.*";
    dialog1.RestoreDirectory = true;
    if (dialog1.ShowDialog() == System.Windows.Forms.DialogResult.OK)
    {
        using (Stream stream = dialog1.OpenFile())
```



```
        {
            bitdecoder = BitmapDecoder.Create(stream,
BitmapCreateOptions.PreservePixelFormat, BitmapCacheOption.OnLoad);
            Img_Alu_Foto.Source = bitdecoder.Frames[0];
            Txt_Alu_Path_Foto.Text = dialog1.FileName;
            Brd_Alu_Foto12.Visibility = Visibility.Collapsed;
            Brd_Alu_Foto.Visibility = Visibility.Visible;
        }
    }
    else
    {
        Img_Alu_Foto.Source = null;
        Brd_Alu_Foto12.Visibility = Visibility.Collapsed;
        Brd_Alu_Foto.Visibility = Visibility.Visible;
    }
}

private void Img_Cerrar_Bus_Alumno_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Brd_Busqueda_Alumno.Visibility = Visibility.Collapsed;
}

private void Btn_Cerrar_Grd_Reg_Alumno_Click(object sender,
System.Windows.RoutedEventArgs e)
{
    Brd_Registro_Alumno.Visibility = Visibility.Collapsed;
}
#endregion

#region Matriculacion
private void Cargar_Matriculacion()
{
    Lbl_FechaMatriculacion.Content = DateTime.Now.ToLongDateString();
    TipoMatricula_Logica Logica_TM = new TipoMatricula_Logica();
    Jornadas_Logica Logica_Jornada = new Jornadas_Logica();
    Especialidad_Logica Logica_Especialidad = new Especialidad_Logica();
    Niveles_Logica Logica_Niveles = new Niveles_Logica();
    Paralelo_Logica Logica_Paralelos = new Paralelo_Logica();
    Periodos_Logica Logica_Periodos = new Periodos_Logica();
    Cbo_Matri_TM.ItemsSource = Logica_TM.Obtener_TipoMatricula();
    Cbo_Matri_Jornada.ItemsSource = Logica_Jornada.Obtener_Jornadas();
    Cbo_Matri_Paralelo.ItemsSource = Logica_Paralelos.Obtener_Paralelo();
    Cbo_Matri_Periodo.ItemsSource = Logica_Periodos.Obtener_PXNombre();
    Cbo_Matri_Nivel.ItemsSource = Logica_Niveles.Obtener_Niveles();
    Cbo_Matri_Especialidad.ItemsSource = Logica_Especialidad.Obtener_Especialidad();
}

private void Cargar_Datos_Matricula()
{
    Matricula_Logica Log_Matricula = new Matricula_Logica();
    Lst_Matricula.ItemsSource = Log_Matricula.Obtener_Matriculas();
    Dgr_Matricula.ItemsSource =
Log_Matricula.Obtener_Matriculas().Select(matr => new
{
    CODIGO = matr.Matri_Codigo,
    FECHA = matr.Matri_Fecha,
    PAGO = matr.Matri_Pago,
    ALUMNO_APELLIDO = matr.Tbl_Alumno.Alu_Apellidos,
    NOMBRE = matr.Tbl_Alumno.Alu_Nombres,
    PERIODO = matr.Tbl_Periodos.Period_Descripcion,
    JORNADA = matr.Tbl_Jornadas.Jor_Nombre,
    ESPECIALIDAD = matr.Tbl_Especialidades.Espe_Nombre,
```



```
        NIVEL = matr.Tbl_Niveles.Niv_Nombre,
        PARALELO = matr.Tbl_Niveles.Niv_Paralelo,
        ESTADO = matr.Matri_Estado
    }).ToList();
    Btn_Nuevo_Matricula.Tag = "";
}
private void Btn_Buscar_AlumnoMatri_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    CargarListadoAlumnosMatricula(Cbo_CriterioBuscarMatri.SelectionBoxItem.ToString(),
    TxtBuscarAlumnoMatri.Text);
}
private void Lst_AlumnosInscripcion1_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Grd_Matriculas.DataContext = Lst_AlumnosInscripcion1.SelectedItem;
}
private void Dgr_ListadoAlumno1_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Lst_AlumnosInscripcion1.SelectedIndex = Dgr_ListadoAlumno1.SelectedIndex;
}
private void Lst_Matricula_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Grd_Matriculas.DataContext = Lst_Matricula.SelectedItem;
}
private void Dgr_Matricula_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Lst_Matricula.SelectedIndex = Dgr_Matricula.SelectedIndex;
}
private void Cbo_Matri_Jornada_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Especialidad_Logica Logica_Especialidad = new Especialidad_Logica();
    try
    {
        Cbo_Matri_Especialidad.ItemsSource =
Logica_Especialidad.ObtenerEspecialidadXJornadas(((Tbl_Jornadas)Cbo_Matri_Jornada.Se
lectedItem).Jor_Codigo);
    }
    catch (Exception)
    {
    }
}
private void Cbo_Matri_Especialidad_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Niveles_Logica Logica_Niveles = new Niveles_Logica();
    try
    {
        Cbo_Matri_Nivel.ItemsSource =
Logica_Niveles.Obtener_NivelesXEspecialidades(((Tbl_Especialidades)Cbo_Matri_Especia
lidad.SelectedItem).Espe_Codigo);
    }
    catch (Exception)
    {
    }
}
```



```
private void Cbo_Matri_Nivel_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
{
    Paralelo_Logica Logica_paralelo=new Paralelo_Logica ();
    try
    {
        Cbo_Matri_Paralelo.ItemsSource =
Logica_paralelo.ObtenerParaleloXNivel(((Tbl_Niveles)Cbo_Matri_Nivel.SelectedItem).Ni
v_Paralelo);
    }
    catch (Exception)
    {
    }
}
private int codigoSEQMatricula()
{
    int seq_CodigoMatricula = 0;
    AcademicoDataContext dc = new AcademicoDataContext();
    try
    {
        seq_CodigoMatricula = dc.Tbl_Matriculas.Max(cod => cod.Matri_Codigo + 1);
    }
    catch (Exception)
    {
        seq_CodigoMatricula = 1;
    }
    return seq_CodigoMatricula;
}
private void Btn_Nuevo_Matricula_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    Tbl_Alumno Alumnos = new Tbl_Alumno();
    if (Lst_AlumnosInscripcion1.SelectedItem != null)
    {
        Lbl_CodigoAlu.Content =
(((Tbl_Alumno)Lst_AlumnosInscripcion1.SelectedItem).Alu_Codigo).ToString();
        string descripcion = string.Format("{0}
{1}",(((Tbl_Alumno)Lst_AlumnosInscripcion1.SelectedItem).Alu_Apellidos),
(((Tbl_Alumno)Lst_AlumnosInscripcion1.SelectedItem).Alu_Nombres));
        Lbl_NomAlumno.Content = descripcion;
    }
    else
    {
        System.Windows.Forms.MessageBox.Show("Seleccione un elemento de la Lista");
        return;
    }
    Cbo_CriterioBuscarMatri.SelectedIndex = -1;
    TxtBuscarAlumno.Text = "";
    Txt_Matri_Codigo.Text = codigoSEQMatricula().ToString();
    Cbo_Matri_TM.SelectedIndex = -1;
    Cbo_Matri_Periodo.SelectedIndex = -1;
    Cbo_Matri_Jornada.SelectedIndex = -1;
    Cbo_Matri_Especialidad.SelectedIndex = -1;
    Cbo_Matri_Nivel.SelectedIndex = -1;
    Cbo_Matri_Paralelo.SelectedIndex = -1;
    Txt_Matri_Pago.Clear();
    Txt_Matri_Alumno.Clear();
    Btn_Nuevo_Matricula.Tag = "Nuevo";}
```



```
private void Btn_Guardar_Matricula_MouseLeftButtonDown(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    if (Cbo_Matri_TM.SelectedItem.ToString() == "")
    {
        System.Windows.Forms.MessageBox.Show("Seleccione El tipo de Matricula");
        return;
    }
    if (Chk_Matri_Pago.IsChecked == false)
    {
        System.Windows.Forms.MessageBox.Show("Verifique el Pago de Matricula");
        return;
    }
    else
    {
        Txt_Matri_Pago.Text = "25";
    }
    if (Btn_Nuevo_Matricula.Tag.ToString() == "Nuevo")
    {
        Tbl_Matriculas NuevaMatricula = new Tbl_Matriculas();
        NuevaMatricula.Matri_Codigo = Convert.ToInt32(Txt_Matri_Codigo.Text);
        NuevaMatricula.Matri_Fecha = DateTime.Now;
        NuevaMatricula.Matri_Pago = Convert.ToDecimal(Txt_Matri_Pago.Text);
        NuevaMatricula.TM_Codigo =
(((Tbl_TipoMatricula)Cbo_Matri_TM.SelectedItem).TM_Codigo);
        NuevaMatricula.Jor_Codigo =
(((Tbl_Jornadas)Cbo_Matri_Jornada.SelectedItem).Jor_Codigo);
        NuevaMatricula.Paralelo_Codigo =
(((Tbl_Paralelo)Cbo_Matri_Paralelo.SelectedItem).Paralelo_Codigo);
        NuevaMatricula.Niv_Codigo =
(((Tbl_Niveles)Cbo_Matri_Nivel.SelectedItem).Niv_Codigo);
        NuevaMatricula.Espe_Codigo =
(((Tbl_Especialidades)Cbo_Matri_Especialidad.SelectedItem).Espe_Codigo);
        NuevaMatricula.Period_Codigo =
(((Tbl_Periodos)Cbo_Matri_Periodo.SelectedItem).Period_Codigo);
        NuevaMatricula.Alu_Codigo = Convert.ToInt32(Lbl_CodigoAlu.Content);
        if (Chk_Matri_Estado.IsChecked == true)
        {
            NuevaMatricula.Matri_Estado = 'A';
        }
        else
        {
            NuevaMatricula.Matri_Estado = 'I';
        }
        if (Lst_Matricula.Items.Count > 0)
        {
            Lst_Matricula.SelectedIndex = 0;
            Matricula_Logica.dc.Refresh(RefreshMode.OverwriteCurrentValues,
Lst_Matricula.SelectedItem);
        }
        Matricula_Logica.dc.Tbl_Matriculas.InsertOnSubmit(NuevaMatricula);
        try
        {
            Matricula_Logica Logica_Matricula = new Matricula_Logica();
            Logica_Matricula.Guardar_Matriculas();
            Cargar_Datos_Matricula();
            string sms = string.Format("El Alumno {0} ha sido Matriculado en
\nEspecialidad: {1}\n,Curso: {2}\n,Paralelo: {3}", Lbl_NomAlumno.Content ,
(((Tbl_Especialidades)Cbo_Matri_Especialidad.SelectedItem).Espe_Nombre),
```



```
((Tbl_Niveles)Cbo_Matri_Nivel.SelectedItem).Niv_Nombre),
(((Tbl_Paralelo)Cbo_Matri_Paralelo.SelectedItem).Paralelo_Descripcion));
        System.Windows.Forms.MessageBox.Show(sms);
    }
    catch (Exception ex)
    {
        System.Windows.Forms.MessageBox.Show("Alumno no
Matriculado\nProceso Terminado" + ex.Message);
    }
}
else
{
    try
    {
        Matricula_Logica Logica_Matricula = new Matricula_Logica();
        Logica_Matricula.Guardar_Matriculas();
        Cargar_Datos_Matricula();
        System.Windows.Forms.MessageBox.Show("Matricula Modificada",
"Matriculacion");
    }
    catch (Exception)
    {
        System.Windows.Forms.MessageBox.Show("Matricula No Modificada",
"Matriculacion");
    }
}
}
private void CargarListadoAlumnosMatricula(string op, string Parametro)
{
    AcademicoDataContext dc = new AcademicoDataContext();
    Alumnos_Logica LogicaAlumnos = new Alumnos_Logica();
    if (Parametro == "")
    {
        System.Windows.Forms.MessageBox.Show("Ingrese al Estudiante");
        return;
    }
    if (op == "")
    {
        System.Windows.Forms.MessageBox.Show("Seleccione el Tipo de
Busqueda");
        return;
    }
    switch (op)
    {
        case "Cedula":

            Lst_AlumnosInscripcion1.ItemsSource =
LogicaAlumnos.Buscar_AlumnoXCedula(Parametro);
            Dgr_ListadoAlumno1.ItemsSource =
LogicaAlumnos.Buscar_AlumnoXCedula(Parametro).Select(car => new
{
                Código = car.Alu_Codigo,
                Apellido = car.Alu_Apellidos,
                Nombres = car.Alu_Nombres,
                Dirección = car.Alu_Direccion,
                Teléfono = car.Alu_Telefono,
                Género = car.Alu_Genero
            }).ToList();
            Btn_Nuevo_Matricula.Tag = "";

```



```
        break;
    case "Nombre Apellido":
        Lst_AlumnosInscripcion1.ItemsSource =
LogicaAlumnos.Buscar_AlumnoXCedula(Parametro);
        Dgr_ListadoAlumno1.ItemsSource =
LogicaAlumnos.Buscar_AlumnoXApellido(Parametro).Select(car => new
        {
            Código = car.Alu_Codigo,
            Apellido = car.Alu_Apellidos,
            Nombres = car.Alu_Nombres,
            Dirección = car.Alu_Direccion,
            Teléfono = car.Alu_Telefono,
            Género = car.Alu_Genero
        }).ToList();
        Btn_Nuevo_Matricula.Tag = "";
        break;
    } }
}
```

Lógica de Matricula

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

namespace Logica_Negocio
{
    public class Matricula_Logica
    {
        public static AcademicoDataContext dc = new AcademicoDataContext();
        public IQueryable<Tbl_Matriculas> Obtener_Matriculas()
        {
            var lista = dc.Tbl_Matriculas.Where(matri => matri.Matri_Estado ==
'A').OrderBy(cod=>cod.Matri_Codigo);
            return lista;
        }
        public IOOrderedQueryable<Tbl_Matriculas>
ObtenerListadosXJor_Esp_Cur_Par(int? Jornada, int? Especialidad, int? curso, int?
Paralelos)
        {
            var listados = dc.Tbl_Matriculas.Where(mat => mat.Jor_Codigo == Jornada &&
mat.Espe_Codigo == Especialidad && mat.Niv_Codigo == curso &&
mat.Paralelo_Codigo == Paralelos && mat.Matri_Estado == 'A').
OrderBy(ord => ord.Tbl_Alumno.Alu_Apellidos);
            return listados;
        }
        public void Guardar_Matriculas()
        {
            try
            {
                dc.SubmitChanges();
            }
            catch (Exception)
            {
                throw;
            }
        }
    }
}
```



Anexo N°14

Cronograma de actividades