

ANÁLISIS DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA WEB PARA EL  
REGISTRO CONTROL Y CONSULTA DE NOTAS PARA LA INSTITUCIÓN EDUCATIVA  
NUEVA VISIÓN DE HONDURAS, UBICADO EN BUENOS AIRES, DEPARTAMENTO  
DEL CAUCA.



EDER GIL RIASCOS RIASCOS

CORPORACIÓN UNIVERSITARIA AUTÓNOMA DEL CAUCA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

2019

ANÁLISIS DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA WEB PARA EL  
REGISTRO CONTROL Y CONSULTA DE NOTAS PARA LA INSTITUCIÓN EDUCATIVA  
NUEVA VISIÓN DE HONDURAS, UBICADO EN BUENOS AIRES, DEPARTAMENTO  
DEL CAUCA.



EDER GIL RIASCOS RIASCOS

TRABAJO DE GRADO PARA OPTAR POR EL TÍTULO DE INGENIERO DE SISTEMAS  
INFORMÁTICOS

MODALIDAD DE PASANTÍA

DIRECTOR DEL PROYECTO  
INGENIERO DE SISTEMAS: GABRIEL ÁNGEL OSORIO

CORPORACIÓN UNIVERSITARIA AUTÓNOMA DEL CAUCA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

2019

## **NOTA DE ACEPTACIÓN**

El trabajo de grado titulado “ANÁLISIS DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA WEB PARA EL REGISTRO CONTROL Y CONSULTA DE NOTAS PARA LA INSTITUCIÓN EDUCATIVA NUEVA VISIÓN DE HONDURAS, UBICADO EN BUENOS AIRES, DEPARTAMENTO DEL CAUCA” realizado por EDER GIL RIASCOS RIASCOS una vez revisado y aprobado para la sustentación, lo autorizan para que realice la gestión administrativa correspondiente para optar por el título de Ingeniero de Sistemas Informáticos.

---

**GABRIEL ÁNGEL OSORIO**  
Director de trabajo

---

**SANTIAGO SÁNCHEZ F.**  
Jurado

---

**CARLOS ANTONIO FLÓREZ A.**  
Jurado

## **DEDICATORIA**

Este trabajo está dedicado principalmente a Dios, quien ha estado presente y acompañándome en esta etapa de mi vida permitiéndome cumplir las metas propuestas hasta el día de hoy. Luego a mis padres, hermanos, familiares y amigos que me han brindado su apoyo total e incondicionalmente hasta el momento.

*Eder Gil Riascos Riascos*

## **AGRADECIMIENTOS**

La agradezco a Dios por haberme acompañado y guiado a lo largo de mi carrera, por brindarme fortaleza en los momentos de debilidad.

Le doy gracias a mis padres, María Enedilde Riascos R. y Oivar Juan Riascos R. por apoyarme en todo momento, por los valores que me han inculcado, por haberme dado la oportunidad de tener una excelente educación en el transcurso de mi vida, y sobre todo por ser los padres que son. Mi ejemplo a seguir.

Le doy gracias a mis hermanos quienes han estado apoyándome en todo momento, con quienes he compartido tristezas y alegrías, quienes me han apoyado en todo momento sin esperar nada a cambio.

Agradezco a demás familiares y amigos que siempre estuvieron cuando más los necesité, con quienes compartí alegrías y tristezas y me brindaron su apoyo constante en la realización de mis metas planteadas.

Le doy gracias al Ingeniero GABRIEL ÁNGEL OSORIO, una persona íntegra y paciente que estuvo dispuesto siempre a brindarme todo su apoyo desde principio a fin de mi carrera universitaria, y durante la realización de este proyecto. Le doy gracias a los profesores que me acompañaron durante el tiempo que estuve estudiando, quienes me brindaron sus conocimientos para mi crecimiento personal y profesional.

## TABLA DE CONTENIDO

1	RESUMEN .....	13
2	ABSTRACT .....	14
3	INTRODUCCIÓN .....	15
4	CAPÍTULO I - PROBLEMA .....	16
4.1	PLANTEAMIENTO DEL PROBLEMA.....	16
4.2	FORMULACIÓN DEL PROBLEMA.....	17
4.3	OBJETIVOS.....	18
4.3.1	OBJETIVO GENERAL .....	18
4.3.2	OBJETIVOS ESPECÍFICOS .....	18
4.4	JUSTIFICACIÓN .....	19
5	CAPÍTULO II - MARCO REFERENCIAL.....	21
5.1	ESTADO DE ARTE O MARCO REFERENCIAL .....	21
5.1.1	ANTECEDENTES .....	21
5.1.2	MARCO TEÓRICO.....	22
5.1.3	GLOSARIO .....	24
6	CAPÍTULO III - METODOLOGÍA .....	26
6.1	METODOLOGÍA .....	26
6.1.1	METODOLOGÍA RUP .....	26
6.2	INGENIERÍA DE SOFTWARE .....	29

6.2.1	DEFINICIÓN .....	29
6.2.2	SISTEMA.....	29
6.2.3	INGENIERÍA DE SISTEMAS .....	30
6.2.4	EL SOFTWARE.....	30
6.2.5	APLICACIONES DEL SOFTWARE .....	31
6.2.6	PROCESO DE DESARROLLO DE SOFTWARE.....	33
6.2.7	MODELOS DE DESARROLLO DE SOFTWARE .....	35
6.2.8	HERRAMIENTAS DE DESARROLLO DE SOFTWARE.....	35
6.2.9	MICROSOFT .NET .....	36
6.2.10	COMPONENTES DE .NET .....	38
6.2.11	BASES DE DATOS .....	43
6.2.12	SQL.....	48
6.2.13	SQL SERVER .....	49
6.2.14	ASP .NET.....	49
7	CAPÍTULO IV - INGENIERÍA DEL PROYECTO .....	51
7.1	DESARROLLO DEL PROYECTO .....	51
7.1.1	FASE DE INICIO.....	51
7.1.2	FASE DE ELABORACIÓN .....	60
7.1.3	FASE DE CONSTRUCCIÓN.....	107
7.1.4	FASE DE TRANSICIÓN.....	111

8	CAPÍTULO V - RESULTADOS .....	112
8.1	RESULTADOS .....	112
9	CAPÍTULO VI - CONCLUSIONES .....	113
9.1	CONCLUSIONES .....	113
9.2	RECOMENDACIONES .....	113
10	BIBLIOGRAFÍA.....	114



## LISTA DE FIGURAS

Ilustración 1. Arquitectura Visual Studio .NET .....	37
Ilustración 2 Visual Studio .NET.....	39
Ilustración 3 Lista para firma de recibido .....	52
Ilustración 4 Planilla de registro y seguimiento académico .....	53
Ilustración 5 Listado de puestos por promedio .....	54
Ilustración 6 Listado de alumnos con asignaturas perdidas .....	55
Ilustración 7 Informe de periodo .....	56
Ilustración 8 Diagrama general de CU– fuente propia.....	61
Ilustración 9 Diagrama de CU Iniciar sesión– fuente propia .....	62
Ilustración 10 Diagrama de CU Administrar docentes– fuente propia .....	62
Ilustración 11 Diagrama de CU Administrar estudiantes– fuente propia .....	62
Ilustración 12 Diagrama de CU Administrar asignaturas– fuente propia .....	63
Ilustración 13 Diagrama de CU Administrar grados– fuente propia .....	63
Ilustración 14 Diagrama de CU Administrar logros– fuente propia.....	63
Ilustración 15 Diagrama de CU Asignar asignaturas– fuente propia.....	64
Ilustración 16 Diagrama de CU Asignar grado– fuente propia .....	64
Ilustración 17 Diagrama CU Administrar nota– fuente propia .....	64
Ilustración 18 Diagrama CU Buscar nota– fuente propia .....	65
Ilustración 19 Diagrama de CU Generar reporte– fuente propia.....	65
Ilustración 20 Diagrama Entidad Relación– fuente propia .....	107
Ilustración 21 IU Iniciar sesión– fuente propia .....	108
Ilustración 22 IU Insertar grados– fuente propia .....	109

Ilustración 23 IU Insertar notas– fuente propia .....	109
Ilustración 24 IU Insertar nota individual– fuente propia .....	110
Ilustración 25 IU Notas estudiante– fuente propia .....	110

## LISTA DE TABLAS

Tabla 1 RF Iniciar sesión.....	57
Tabla 2 RF Administrador.....	58
Tabla 3 RF Docentes.....	59
Tabla 4 RF Estudiantes .....	59
Tabla 5 RNF Sistema .....	59
Tabla 6 CU Iniciar sesión.....	65
Tabla 7 CU Nuevo docente .....	66
Tabla 8 CU Eliminar docente .....	67
Tabla 9 CU Actualizar docente.....	69
Tabla 10 CU Buscar docente .....	70
Tabla 11 CU Asignar asignaturas .....	71
Tabla 12 CU Nuevo estudiante.....	73
Tabla 13 CU Eliminar estudiante.....	74
Tabla 14 CU Actualizar estudiante .....	75
Tabla 15 CU Buscar estudiante .....	77
Tabla 16 CU Asignar grado.....	78
Tabla 17 CU Nueva asignatura.....	79
Tabla 18 CU Eliminar asignatura .....	80
Tabla 19 CU Actualizar asignatura.....	81
Tabla 20 CU Buscar asignatura .....	83
Tabla 21 CU Nuevo grupo .....	84
Tabla 22 CU Eliminar grupo .....	85

Tabla 23 CU Nuevo logro .....	86
Tabla 24 CU Eliminar logro .....	87
Tabla 25 CU Actualizar logro.....	88
Tabla 26 CU Buscar logro .....	90
Tabla 27 CU Registrar notas .....	91
Tabla 28 CU Ver notas.....	93
Tabla 29 CU Generar reportes.....	94
Tabla 30 - Tabla Docente .....	96
Tabla 31 - Tabla Logros .....	97
Tabla 32 - Tabla Docente Grupo .....	98
Tabla 33 - Tabla Asignatura .....	99
Tabla 34 - Tabla Grupo .....	100
Tabla 35 - Tabla Grado .....	101
Tabla 36 - Tabla Notas .....	101
Tabla 37 - Tabla Estudiante Grupo.....	102
Tabla 38 - Tabla Estudiante.....	103
Tabla 39 - Tabla Periodo .....	105
Tabla 40 Tabla Tipo usuario.....	106

## 1 RESUMEN

Este trabajo fue realizado con el objetivo de sistematizar el proceso de control, registro y consulta de las notas en la Institución Educativa Nueva Visión de Honduras, utilizando como guía de desarrollo la metodología RUP. Logrando obtener una aplicación que garantice el tratamiento, integridad y portabilidad de la información mediante el uso de una interfaz agradable que facilite el manejo de la aplicación, sin olvidar la generación de informes que permitan medir el desempeño de los estudiantes en esta institución educativa.

Para el desarrollo de este trabajo se utilizó el IDE (Entorno de desarrollo integrado) de Visual Studio, utilizando como lenguaje de programación C#, y el desarrollo de la base de datos mediante SQL Server Manager Studio 17 en su versión Express.

**Palabras claves:** Notas, Control, Registro, Metodología RUP, Informes, Tratamiento, Integridad, Portabilidad

## 2 ABSTRACT

This work was carried out with the objective of systematizing the process of control, registration and consultation of the notes in the Institución Educativa Nueva Visión de Honduras, using the RUP methodology as a development guide. Achieving an application that guarantees the treatment, integrity and portability of the information through the use of an interface that guarantees the easy handling of the application, without forgetting the management of reports that allow measuring the performance of students in this educational institution.

For the development of this work we used the IDE (Integrated Development Environment) of Visual Studio, using the C # programming language, and the development of the database using SQL Server Manager Studio 17 in its Express version.

Key words: Notes, Control, Registration, RUP Methodology, Reports, Treatment, Integrity, Portability

### 3 INTRODUCCIÓN

Hoy, más que nunca es de vital importancia pensar en el uso apropiado que se le debe dar a las tecnologías de la información en los procesos de enseñanza y sistematización de los datos, llevando un control automatizado y efectivo sobre el registro de datos, los cuales garanticen su adecuado tratamiento, optima integridad y portabilidad de la información.

Debido a esto, la finalidad de este trabajo consiste en crear una aplicación web para el registro, control y consulta de notas que será implementado en la Institución Educativa Nueva Visión de Honduras, Corregimiento de Honduras – Buenos aires, Departamento del Cauca. Este trabajo se desarrolla mediante la utilización de la metodología RUP, esta se basa en la utilización de una serie de CU (casos de uso), los cuales son la guía principal para el desarrollo e implementación de la aplicación, estos se deben ir desarrollando de una forma iterativa e incremental.

Para el cumplimiento del proyecto se pretende utilizar las versiones EXPRES de: Microsoft Visual Studio .Net 2017 y SQL Server v 17. Realizando el desarrollo del código mediante el lenguaje de programación de C# (C Sharp).

## **4 CAPÍTULO I - PROBLEMA**

### **4.1 PLANTEAMIENTO DEL PROBLEMA**

El alto desarrollo tecnológico que se vive en la actualidad provoca que toda empresa, compañía o entidad, sin importar su tipo, deba contar con herramientas tecnológicas, que ofrezcan productos y/o servicios que satisfagan las necesidades de la comunidad, permitiéndoles estar a la vanguardia en el uso de estas nuevas tecnologías, presentando productos y/o servicios de mejor calidad.

La Institución Educativa Nueva Visión de Honduras, Buenos Aires, Cauca, en la parte administrativa y de control, desde hace mucho tiempo, lleva los registros de notas de los estudiantes de forma manual, haciéndolo a través de planillas impresas y después diligenciadas a mano por los docentes, para posteriormente enviarlas a terceros para transcribirlas y luego imprimirlas para presentar informes de las notas a los padres de familia y dejar informe de estas notas en la institución educativa, archivándolas en carpetas, las cuales con el tiempo se deterioran y por ende, no son seguras.

El actual manejo del registro de notas en esta institución educativa, conlleva una serie de inconvenientes como errores en la transcripción, mayor inversión de tiempo en la búsqueda de estas notas, baja confiabilidad en la información, mayor tiempo de procesamiento, y maximizan los costos.

Al llevar el registro de las notas en archivos impresos y almacenados en carpetas, con el tiempo éstos registros se pueden destruir o alterar en su contenido, causando una pérdida de la



información, viéndose afectado el registro histórico de las notas de los estudiantes, afectando a estudiantes y exalumnos en la solicitud de constancias de notas para estudios posteriores.

#### **4.2 FORMULACIÓN DEL PROBLEMA.**

¿Llevando a cabo, el análisis diseño e implementación de una plataforma web para el registro control y consulta de notas para la Institución Educativa Nueva Visión de Honduras, Municipio de Buenos Aires, Departamento del Cauca, se contribuirá a elevar la calidad tecnológica de la Institución Educativa generando un impacto social positivo a la comunidad?

## **4.3 OBJETIVOS**

### **4.3.1 OBJETIVO GENERAL**

- Analizar, diseñar e implementar una plataforma web para el registro control y consulta de notas para la Institución Educativa Nueva Visión de Honduras, Municipio de Buenos Aires, Departamento del Cauca

### **4.3.2 OBJETIVOS ESPECÍFICOS**

- Brindar a la comunidad educativa un sistema que permita administrar las notas de los estudiantes.
- Sistematizar el proceso de control, registro y consulta de notas de los estudiantes que realizan los docentes y administrativos.
- Brindar una herramienta con la cual se puedan realizar copias de seguridad y proporcione la portabilidad de la información.
- Ofrecer a la comunidad educativa la posibilidad de consultar la información a cualquier hora y desde cualquier lugar.

#### 4.4 JUSTIFICACIÓN

El registro, control y consulta de notas forma parte de los libros reglamentarios que los docentes deben diligenciar, estableciéndose como soporte o evidencia de la gestión académica de las instituciones educativas; estos registros se convierten en históricos que dan cuenta de la operatividad en uno de los procesos que desarrolla una institución educativa, facilitando la consulta de acciones y toma de decisiones en estas instituciones.

Es debido a esto, y de acuerdo a la problemática planteada, que la Institución Educativa Nueva Visión de Honduras, ubicada en el Municipio de Buenos Aires, del Departamento del Cauca, debe aprovechar el desarrollo tecnológico para ofrecer un mejor servicio a la comunidad educativa sistematizando el registro control y consulta de notas de los estudiantes mediante un sitio web, permitiendo a los docentes llevar un control ágil, fácil y seguro de las calificaciones de los estudiantes, como también elaborar estadísticas de las mismas, además de permitir que los padres de familia puedan saber cómo van sus hijos, y los estudiantes puedan visualizar en cualquier momento sus respectivas calificaciones.

Hoy en día hay muchos sistemas para el registro control y consulta de notas, pero no son lo suficientemente flexibles que se adapten a las necesidades de la institución educativa, además de sus altos costos, es por ello que la Institución Educativa Nueva Visión de Honduras, ubicada en Buenos Aires, Cauca, se ve en la necesidad de contar con su propio sistema de acuerdo a la flexibilidad y características requeridas por ellos mismos.

Mediante el uso del sistema de registro, control y consulta de notas se estarían agilizando estos procesos, como también, se garantizaría la integridad de esta información. Se estaría utilizando una base de datos al beneficio de todos los actores participantes en la comunidad educativa.

“El registro control y consulta de notas se encuentra establecido en los libros reglamentarios de los docentes que se encuentra estipulado en el Decreto 1860 de 1994 (agosto 3) Ministerio de Educación Nacional”<sup>1</sup>

La Ley 594 de 2000 o Ley General de Archivo, en especial su Artículo 3° y el Decreto 1290 de 2008.

---

<sup>1</sup> COLOMBIA. Ministerio de Educación Nacional. Decreto 1860 de 1994 (agosto 3). “Por el cual se reglamenta parcialmente la Ley 115 de 1994, en los aspectos pedagógicos y organizativos generales”. Diario oficial No 41.473, del 5 de agosto de 1994

## **5 CAPÍTULO II - MARCO REFERENCIAL**

### **5.1 ESTADO DE ARTE O MARCO REFERENCIAL**

#### **5.1.1 ANTECEDENTES**

En un proyecto realizado por David Meza, para optar por el título de Ingeniero de Sistemas; investigo, diseño y construyó un sistema de control para las calificaciones en la Escuela Politécnica del Ejército (ESPE), resaltando la importancia del uso de las tecnologías basadas en internet para satisfacer las necesidades de las instituciones, de hacer más eficientes los procesos académicos, para de esta manera simplificar las labores cotidianas.

Otro trabajo investigativo, llevó a cabo el desarrollo de un sitio web, para un colegio público, con el objetivo de brindar información acerca del colegio a usuarios no registrados y proporcionar una serie de funcionalidades a los alumnos y profesores del centro educativo. Estos trabajos investigativos se relacionan con el que se está proponiendo, para que toda institución educativa, eleve su calidad tanto administrativa como tecnológica, porque la tecnología en el mundo actual, es vista como una parte integral de toda actividad de conocimiento.

### 5.1.2 MARCO TEÓRICO

#### **School Pack:**

“Un sistema que te ayuda a coordinar, programar y evaluar los procesos educativos de tu Institución. Con una amplia gama de servicios, que permiten centralizar los procedimientos, administrar las tareas e implementar un alto nivel de seguridad sobre la información de tu sistema educativo”<sup>2</sup>.

#### **Colweb Colweb- Sistema de Gestión Académica.**

“Esta herramienta “Colweb” se ha convertido en la más eficaz solución para la **administración de la gestión académica de los colegios**, que facilita el cumplimiento de los procesos administrativos exigidos por el decreto 1290/2009, la información solicitada por la ley 1620, la socialización y seguimiento del PEI, la publicación del plan de estudio y del sistema de evaluación institucional, logrando:

- Mejorar la comunicación de toda la comunidad educativa.
- Intensificar el seguimiento a los estudiantes Una significativa disminución de costos y de esfuerzos administrativos.
- Disminuir a cero los tiempos para entrega de boletines y demás informes académicos.
- Reducir significativamente la posibilidad de errores.
- Herramienta de apoyo para el control y seguimiento de los sistemas de calidad”<sup>3</sup>.

#### **Ciudad Educativa**

“Automatiza el reporte de logros, estándares y nota definitiva para reducir las tareas repetitivas del maestro, quien ahora puede reutilizar actividades, usar las plantillas online, calificar con el

---

<sup>2</sup> SHOOL PACK. Sistema de información integral para instituciones educativas. Cundinamarca, Colombia. 2000. Disponible en: [http://www.upwaresoft.com/flipbookSchool/pages/school\\_pack.pdf](http://www.upwaresoft.com/flipbookSchool/pages/school_pack.pdf).

<sup>3</sup> COLWEB. Sistema de Gestión Académica. Disponible en: <http://www.colweb.com.co/nosotros.html>

micrófono o utilizar un archivo Excel si no hay internet disponible. Además, puede compartir videos o documentos, crear foros y chatear con papás y estudiantes”<sup>4</sup>.

Es importante mencionar que en la actualidad hay mucho software que permite realizar labores similares a la del producto a realizar, pero como sustento para el desarrollo de este proyecto, es que, este se desarrolla teniendo en cuenta las necesidades exclusivas de la Institución Educativa Nueva Visión de Honduras, acorde a los requerimientos esenciales de los directivos de esta Institución Educativa, además de contar con una gran adaptabilidad, integridad de la información, portabilidad y fácil utilización de la aplicación, quedando disponible para posibles mejoras y/o adaptación de nuevas funcionalidades que se deseen implementar.

Además de mencionar que la mayoría de software está diseñado para ser usado en una población en general, contando con funcionalidades limitadas y/o en ocasiones muy complicadas para su utilización, por otro lado, otro tipo de software se desarrolla para un usuario en particular el cual se adapta a sus necesidades propias cumpliendo los requerimientos y/o características que el usuario en cuestión requiere. Es debido a esto que este software se adapta exclusivamente a las necesidades y requerimientos de la Institución Educativa Nueva Visión de Honduras.

---

<sup>4</sup> CIUDAD EDUCATIVA. Portafolio 2018. Disponible en: <https://www.ciudadeducativa.com/ciudadeducativa-colegio.pdf>

### 5.1.2.1 *CONTEXTO*

Este trabajo se realiza en la Institución Educativa Nueva Visión de Honduras, ubicada en el Corregimiento de Honduras – Buenos aires, Departamento del Cauca

**Área:** Rural

Esta Institución Educativa es de carácter pública, y se labora en la jornada de la mañana.

Dicha Institución Educativa se encuentra dividida en dos sedes:

- Sede 1: presenta los siguientes niveles educativos Básica secundaria y Media conformada por 15 grupos, 20 docentes y aproximadamente 425 estudiantes
- Sede 2: presenta los siguientes niveles educativos: Preescolar y Básica primaria, conformada por 11 grupos, 11 docentes y aproximadamente 200 estudiantes

### 5.1.3 **GLOSARIO**

**RUP:** (El Proceso Unificado Racional) es un proceso de ingeniería del software. Proporciona un acercamiento disciplinado a la asignación de tareas y responsabilidades en una organización de desarrollo<sup>5</sup>.

**IDE:** (Entorno de desarrollo integrado) “es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI)”<sup>6</sup>.

**Casos de uso (CU):** Los casos de uso son una técnica para especificar el comportamiento de un sistema<sup>7</sup>.

---

<sup>5</sup> MARTÍNEZ, Alejandro y MARTÍNEZ, Raúl. Guía a Rational Unified Process.

<sup>6</sup> FERGACIAC. Entorno de desarrollo integrado IDE. Programación orientada a objetos. 2013. Disponible en: <https://fergaciac.wordpress.com/2013/01/25/entorno-de-desarrollo-integrado-ide/>

<sup>7</sup> ARIAS CHAVES, Michael. La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software



**UML:** Es un estándar OMG diseñado para visualizar, especificar, construir y documentar software orientado a objetos<sup>8</sup>.

**HTML:** Lenguaje de Marcado para Hipertextos (HyperText Markup Language) es el elemento de construcción más básico de una página web y se usa para crear y representar visualmente una página web<sup>9</sup>.

**WEB:** La World Wide Web o simplemente WWW o Web es uno de los métodos más importantes de comunicación que existe en Internet. Consiste en un sistema de información basado en Hipertexto (texto que contiene enlaces a otras secciones del documento o a otros documentos). La información reside en forma de páginas Web en ordenadores que se denominan servidores Web y que forman los nodos de esta telaraña. Se denomina páginas Web a documentos que contienen elementos multimedia (imágenes, texto, audio, vídeo, etc.) además de enlaces de hipertexto.

**CSS:** Es un lenguaje de diseño gráfico para definir y crear la presentación de un documento.

**FRAMEWORKS:** Es un esquema para el desarrollo y/o la implementación de una aplicación<sup>10</sup>.

**RF:** (Requerimiento funcional), es la descripción que informa sobre el comportamiento del software.

**RNF:** (Requerimiento no funcional), son herramientas que se requieren para el buen funcionamiento del software, o su desarrollo.

**PEI:** (Proyecto educativo institucional), es un documento para escuelas y colegios, donde se establecen los principios y fines del establecimiento, los recursos docentes y didácticos disponibles y necesarios, la estrategia pedagógica, el reglamento para docentes y estudiantes y el sistema de gestión escolar.

---

<sup>8</sup> BOOCH, Graby; RUMBAUGH, Jim, JACOBSON, Ivar. El lenguaje unificado de modelado.

<sup>9</sup> MDN web docs. HTML. 2017. Disponible en: <https://developer.mozilla.org/es/docs/Web/HTML>

<sup>10</sup> JORDISAN.NET. ¿Qué es un framework? Disponible en: <https://jordisan.net/blog/2006/que-es-un-framework/>

## **6 CAPÍTULO III - METODOLOGÍA**

### **6.1 METODOLOGÍA**

Se podrían considerar las metodologías de desarrollo de software como una guía de trabajo mediante la cual se estructuran y planifican los diferentes procesos de desarrollo; en la actualidad se cuenta con una gran variedad de estas metodologías las cuales tienen sus propias características, ofreciendo ventajas y desventajas en su uso e implementación con respecto a otras, pero queda a criterio del desarrollador o equipo de trabajo elegir con cuál de estas metodologías trabajar. Es debido a esto que para el desarrollo de este trabajo se seleccionó la metodología de desarrollo RUP, ya que es adaptable a posibles cambios sin afectar en gran medida el desarrollo realizado.

#### **6.1.1 METODOLOGÍA RUP**

“El Rational Unified Process o Proceso Unificado de Racional. Es un proceso de ingeniería de software que suministra un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta y de mayor calidad para satisfacer las necesidades de los usuarios que tienen un cumplimiento al final dentro de un límite de tiempo y presupuesto previsible. Es una metodología de desarrollo iterativa que está enfocada hacia “diagramas de los casos de uso, manejo de los riesgos y el manejo de la arquitectura” como tal.

El RUP mejora la productividad del equipo de trabajo ya que permite que cada miembro del grupo sin importar su responsabilidad específica pueda acceder a la misma base de datos incluyendo sus

conocimientos. Esto hace que todos compartan el mismo lenguaje, la misma visión y el mismo proceso acerca de cómo desarrollar un software”<sup>11</sup>.

## **Fases**

**Fase de inicio:** El objetivo de esta fase es la comunicación, realizar el modelado de negocio y la captura de los requerimientos de la Institución Educativa Nueva Visión de Honduras. Esto se pretende realizar mediante el uso y aplicación de:

- Entrevista
- Casos de uso
- Lluvia de ideas
- Prototipos

Analizar la información y escribir los requerimientos esenciales (funcionalidad rendimiento y restricciones que debe tener el producto) mediante las iteraciones que considere necesarias.

**Fase de elaboración:** El objetivo principal de esta fase es crear los diagramas de: clases de análisis y clases de diseño, además de identificar los riesgos claves. Para lograr óptima satisfacción de esto las actividades a realizar durante este desarrollo son: el modelado de negocio, realizar el análisis y diseño de la aplicación e implementar algunos de los casos de uso más críticos del sistema.

**Fase de construcción:** El objetivo de esta fase es la construcción del producto mediante una serie de iteraciones, realizando las siguientes actividades:

---

<sup>11</sup> MAGAZINE. Metodología RUP y Ciclo de Vida. Disponible en: <http://rupmetodologia.blogspot.com/>

- Seleccionando y desarrollando los casos de uso restantes
- Realizar el diseño del sistema
- Realizar la programación necesaria
- Integrar todas las partes del sistema
- Documentar las actividades realizadas
- Realizar las pruebas necesarias.

**Fase de transición:** En esta fase el sistema software se entrega al usuario final para sus respectivas pruebas en un entorno real. Al terminar esta fase se debe tener un software documentado y funcionando correctamente.

- Se realiza la capacitación al usuario
- Ejecutan pruebas finales (garanticen el producto)
- Se realiza el cierre de la documentación

La metodología RUP es un proceso iterativo e incremental que permite la adaptabilidad a cambios durante el proceso de desarrollo, donde el trabajo se divide en partes más pequeñas o mini proyectos seleccionando algunos casos de uso y realizarlos durante cada una de las iteraciones que se realizan en el desarrollo del proyecto, permitiendo un equilibrio entre casos de uso y arquitectura.

## 6.2 INGENIERÍA DE SOFTWARE

A continuación, se dará a conocer una visión general sobre la ingeniería de software con el propósito de dar al lector un pequeño enfoque sobre las generalidades del desarrollo de software, definiendo algunos conceptos técnicos y fundamentales que se trabajan durante el desarrollo de software.

### 6.2.1 DEFINICIÓN

“Ingeniería del Software es el estudio de los principios y metodologías para desarrollo y mantenimiento de sistemas de software. [Zelkovitz, 1978]”<sup>12</sup>. “Es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como desarrollo de software o producción de software. [Bohem, 1976]”. “Es una disciplina que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de éste después de que se utiliza. [Sommerville, 2001]”

### 6.2.2 SISTEMA

“Un sistema es una colección de componentes interrelacionados que trabajan conjuntamente para cumplir algún objetivo”<sup>13</sup>

---

<sup>12</sup> BAYONA ZUBIETA, Jenny Carolina, PINEDA SAMACÁ, Olga Lucía y PARDO MAHECHA, Oscar Daniel. El papel de la Ingeniería de Software en el desarrollo de aplicaciones.

<sup>13</sup> SOMMERVILLE Ian. Ingeniería de software

### **6.2.3 INGENIERÍA DE SISTEMAS**

“La ingeniería de sistemas consiste en la actividad de especificar, diseñar, implementar, validar, distribuir y mantener sistemas como un todo. Los ingenieros de sistemas no sólo están relacionados con el software, sino también con el hardware y las interacciones del sistema con los usuarios y su entorno de trabajo”<sup>14</sup>.

### **6.2.4 EL SOFTWARE**

Es un conjunto de “Instrucciones (programas de cómputo) que cuando se ejecutan proporcionan las características, función y desempeño buscados; 2) estructuras de datos que permiten que los programas manipulen en forma adecuada la información, y 3) información descriptiva tanto en papel como en formas virtuales que describen la operación y uso de los programas”<sup>15</sup>.

Según Roger S. Pressman en su libro “Ingeniería de software un enfoque práctico” dice que el software de computadora es un producto al que se le da mantenimiento durante mucho tiempo, programas que se ejecutan en una computadora de cualquier tamaño y arquitectura, contenido que se presenta a medida que se ejecutan los programas de cómputo. Estos programas son realizados por ingenieros de software, quienes también son encargados de darle el mantenimiento pertinente.

En la actualidad la principal importancia del software es que afectan en casi todos los aspectos de la vida de las personas, además de haber invadido el comercio y actividades cotidianas. mediante

---

14 OLGUIN ESPINOZA, José Martín. Análisis orientado a objetos. 2004. Disponible en:  
<http://yaqui.mx/l.uabc.mx/~molguin/as/IngSoft%201-4.pdf>

15 PRESSMAN S. Roger. Ingeniería de software: un enfoque practico. Unidad 1. Disponible en:  
<http://www.itlalaguna.edu.mx/academico/carreras/sistemas/ingsoftware1/Unidad1.pdf>

el uso de la ingeniería de software se pueden construir sistemas complejos en breves periodos de tiempo y de alta calidad.

Con el propósito de obtener un producto exitoso, lo recomendable es usar un proceso ágil y adaptable, obteniendo resultados de calidad, los cuales deben satisfacer las necesidades del usuario, adquiriendo como producto final desde el punto de vista del ingeniero de software, un conjunto de programas, documentación y otros que constituye el software, pero desde el punto de vista del cliente el producto final es la información y la generación de los informes que de algún modo le facilita las tareas que realiza, mejorando el mundo en el que vive.

**Características del software:** es un elemento de un sistema lógico y no físico, según esto tiene sus propias características:

- El software se desarrolla o modifica; no se manufactura en el sentido clásico
- El software no se “desgasta”.
- El software se desarrolla a medida de las necesidades del cliente.

### **6.2.5 APLICACIONES DEL SOFTWARE**

En la actualidad el software de computadora se puede agrupar en siete grandes categorías, generando retos continuos a los ingenieros de software:

**Software de sistemas:** Conjunto de programas informáticos que ofrecen servicios a otros programas. (Por ejemplo: compiladores, editores, herramientas para administrar archivos,

componentes de sistemas operativos, software de redes) por lo general este tipo de software tiene gran interacción con el hardware de la computadora.

**Software de aplicación:** Son programas que resuelven una necesidad específica de negocio, estas aplicaciones procesan datos comerciales facilitando las operaciones de negocio o la toma de decisiones administrativas, controlan funciones de negocio en tiempo real procesamiento de transacciones en punto de venta.

**Software de ingeniería y ciencias:** Son algoritmos diseñados especialmente para el procesamiento de una gran cantidad de números

**Software incrustado:** Es el software que reside dentro de un producto o sistema, y sirve para implementar una función específica o controlar algunas características y/o funciones para el usuario

**Software de línea de productos:** Este tipo de software está enfocado principalmente para el uso de muchos consumidores diferentes, se centra en algún mercado limitado y particular (ejemplo: control de inventario) o se dirige a una población mayor de consumidores (Procesamiento de datos, multimedia, entretenimiento, base de datos entre otras)

**Aplicaciones web:** También llamadas “WEBAPPS” se centra principalmente en las redes y agrupa una gran cantidad de aplicaciones, en un principio eran un conjunto de archivos de hipertexto vinculados entre sí que presentaban información, pero desde la web 2.0 en adelante estas



aplicaciones web evolucionaron a tal punto que se pueden integrar a bases de datos corporativas y aplicaciones de negocios.

**Software de inteligencia artificial:** Esta área se enfoca en algoritmos no numéricos para resolver problemas complejos, las aplicaciones de esta área incluyen. robótica, reconocimiento de patrones (imagen, voz), redes neuronales, juegos entre otros.

### **6.2.6 PROCESO DE DESARROLLO DE SOFTWARE**

Tiene como propósito lograr un producto software eficaz y eficiente que cumpla con los requisitos del cliente.

“Un producto software en sí es complejo, es prácticamente inviable conseguir un 100% de confiabilidad de un programa por pequeño que sea. Existe una inmensa combinación de factores que impiden una verificación exhaustiva de todas las posibles situaciones de ejecución que se puedan presentar (entradas, valores de variables, datos almacenados, software del sistema, otras aplicaciones que intervienen, el hardware sobre el cual se ejecuta, etc.). Un producto software es intangible y por lo general muy abstracto, esto dificulta la definición del producto y sus requisitos, sobre todo cuando no se tiene precedentes en productos software similares. Esto hace que los requisitos sean difíciles de consolidar tempranamente. Así, los cambios en los requisitos son inevitables, no sólo después de entregado en producto sino también durante el proceso de desarrollo”<sup>16</sup>.

---

<sup>16</sup> LETELIER, P. Proceso de desarrollo de software.

El proceso de desarrollo de software se define dentro de cuatro actividades fundamentales, las cuales son:

- **Especificaciones del software:** Funcionalidad y restricciones
- **Desarrollo del software:** Diseña y construye el software
- **Validación del software:** Se realizan las validaciones pertinentes
- **Evolución del software:** Se adapta a las necesidades del cliente

“Un proceso de desarrollo de software es la descripción de una secuencia de actividades que deben ser seguidas por un equipo de trabajadores para generar un conjunto coherente de productos, uno de los cuales es el programa del sistema deseado”<sup>17</sup>.

El objetivo básico del proceso es hacer predecible el trabajo que se requiere:

- Predecir el costo.
- Mantener un nivel de calidad
- Predecir el tiempo de desarrollo

Otro enfoque en el desarrollo de software es establecer las relaciones entre los elementos que intervienen: **Quién, Qué, Cuándo, y Cómo.**

**Quién:** Las personas que participan en el proyecto y sus roles

**Qué:** El objeto que se desea realizar

**Cuándo y Cómo:** Las actividades que se deben ir realizando durante el desarrollo del proyecto

---

<sup>17</sup> DRAKE, J. M. Programación orientada a objetos: Lenguajes, Metodologías y Herramientas Master de Computación.

Con el fin de lograr un óptimo desarrollo de software se han creado varios modelos a seguir, cada uno de los cuales describe un enfoque diferente con el uso e implementación de diferentes actividades.

### **6.2.7 MODELOS DE DESARROLLO DE SOFTWARE**

Se define como una secuencia de pasos a seguir para la representación simplificada de un proceso de software:

- Codificar y corregir
- Modelo en cascada
- Desarrollo evolutivo
- Desarrollo formal de sistemas
- Desarrollo basado en reutilización
- Desarrollo incremental
- Desarrollo en espiral

### **6.2.8 HERRAMIENTAS DE DESARROLLO DE SOFTWARE**

Son herramientas informáticas que ofrecen una gran ayuda en el proceso de ciclo de vida de software, es utilizada por el ingeniero de software para Crear, depurar, Gestionar o mantener un programa informático. Software que se usa para ayudar en las actividades de software (Diseño e implementación de nuevo software).

“Estas herramientas tienen como finalidad, el disminuir el estrés y los tiempos de cada fase, para además mejorar los resultados obtenidos y dar mejores propuestas al cliente”<sup>18</sup>. Mediante el uso de estas herramientas resulta mucho más fácil la implementación y/o desarrollo del producto que satisfaga las necesidades del cliente, listando a continuación el tipo de herramienta que se deben utilizar en cada una de las fases que conlleva el proceso de desarrollo de software:

- Herramientas para Modelado de Software
- Herramientas de Desarrollo de Software
- Herramientas para Hacer Pruebas del Desarrollo de SW
- Herramientas para Depurar Código
- Herramientas para el Diseño de Software

### **6.2.9 MICROSOFT .NET**

La Plataforma .NET hace referencia a la tecnología de programación de .NET Framework, es una herramienta para la creación de aplicaciones. Para esto Microsoft, cuenta con su propio IDE de desarrollo el cual integra una gran cantidad de funcionalidades que la ubican en la cima del desarrollo de aplicaciones, junto con JAVA y su arquitectura J2EE.

“Microsoft .NET Framework o abiertamente .NET se trata de un entorno de desarrollo multilenguaje diseñado por Microsoft para simplificar la construcción, distribución y ejecución de aplicaciones para internet. Tiene fundamentalmente tres componentes: una máquina virtual (CLR: Common Language Runtime) que procesa código escrito en un lenguaje intermedio (MSIL:

---

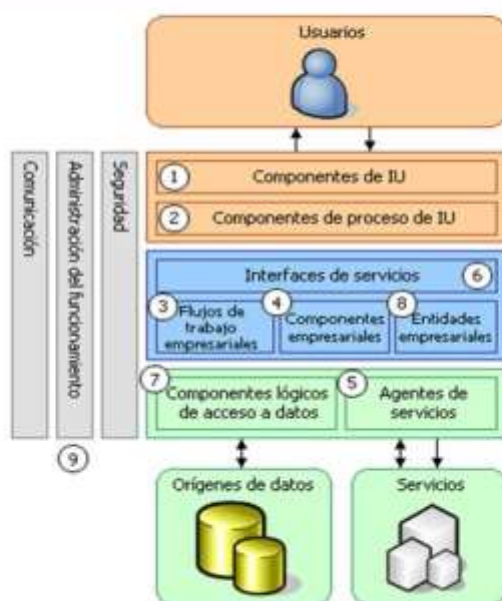
<sup>18</sup> OK HOSTING. Herramientas de desarrollo de software. Disponible en: <https://okhosting.com/blog/herramientas-de-desarrollo-de-software/>

Microsoft Intermediate Language), una biblioteca de clases (biblioteca .NET) a ASP.NET que proporciona los servicios necesarios para crear aplicaciones web”<sup>19</sup>.

Además, .NET incluye dentro de su tecnología un análisis y diseño los cuales permiten hacer uso de mejores técnicas de desarrollo de aplicaciones.

Niveles de desarrollo de aplicaciones .NET

## Arquitectura Visual Studio .Net



Capas de componentes de servicios y aplicaciones distribuidas creadas con .NET  
<http://www.microsoft.com/>

Ilustración 1. *Arquitectura Visual Studio .NET*  
*Modelo de arquitectura de aplicaciones .NET*

<sup>19</sup> CEBALLOS, Francisco Javier. Microsoft Visual Basic .NET Lenguaje y aplicaciones. 3ª ed.

Referenciando la imagen anterior (Modelo de arquitectura de aplicaciones .NET) .NET brinda distintos niveles de aplicaciones, ofreciendo la posibilidad de dividir la arquitectura de una aplicación en capas.

**Capa de presentación:** Hace referencia a la presentación o diseño del programa ante el usuario final, debe manejar interfaces que cumplan con la lógica que ésta maneja, además de facilitar al usuario la interacción con la aplicación. Dar cumplimiento de manera agradable a las necesidades del usuario, la interfaz debe ser amigable y fácil de utilizar, ya que el usuario final es el encargado de utilizar el sistema.

**Capa de negocios:** Es aquí donde se encuentra toda la lógica del programa, así como la encargada de comunicar los datos de la aplicación con la interfaz de usuario, conectar al usuario final con los datos que maneja la aplicación, las estructuras de datos y objetos encargados para la manipulación de los datos existentes, así como el procesamiento de la información ingresada o solicitada por el usuario en la capa de presentación. Es aquí donde se definen las reglas que debe cumplir la aplicación.

**Capa de acceso a datos:** Es la encargada de realizar las transacciones con la base de datos, obtiene la información de la base de datos y la envía a la capa lógica si es el caso o viceversa.

#### **6.2.10 COMPONENTES DE .NET**

Microsoft .NET está compuesto por:

### 6.2.10.1 Plataforma .NET

Tiene como objetivo principal, simplificar el desarrollo de aplicaciones web, y se encuentra soportado sobre los estándares de los servicios web; esta plataforma se encuentra diseñada para permitir el desarrollo de software mediante la mayoría de los lenguajes de programación, siempre y cuando cumpla con las normas estipuladas por la plataforma .NET.

Es importante mencionar que para crear aplicaciones mediante el uso de la plataforma .NET en sus diferentes áreas como lo son: servicios web o aplicaciones tradicionales (aplicaciones de consola, aplicaciones de ventanas, servicios de Windows NT, etc.), Microsoft cuenta con un kit de desarrollo conocido como .NET Framework. El cual contiene el CLR (Common Language Runtime), .NET Framework Clases y características avanzadas como ADO.NET (para acceso a bases de datos), ASP.NET (para generar páginas activas) y WinForms (para construir aplicaciones Windows). Todo esto se puede realizar mediante el uso de Visual Studio, que cuenta con una interfaz agradable y fácil de utilizar.

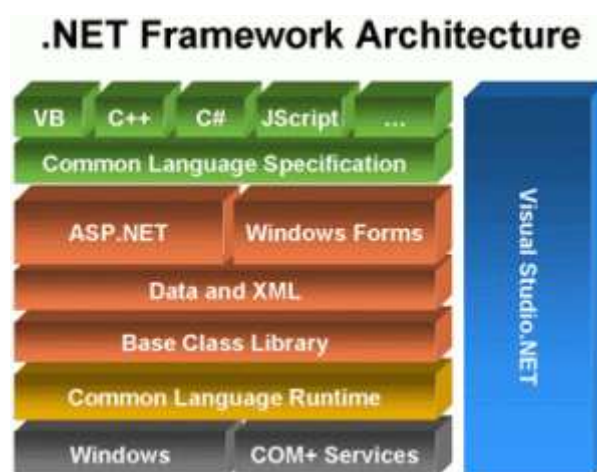


Ilustración 2 Visual Studio .NET

Tomada de: <http://www.bestdotnettraininginchennai.com/components-of-net-architecture/>

### **6.2.10.2 .NET Framework SDK**

“.NET Framework es un entorno de ejecución administrado para Windows que proporciona diversos servicios a las aplicaciones en ejecución. Consta de dos componentes principales: Common Language Runtime (CLR), que es el motor de ejecución que controla las aplicaciones en ejecución, y la biblioteca de clases de .NET Framework, que proporciona una biblioteca de código probado y reutilizable al que pueden llamar los desarrolladores desde sus propias aplicaciones. Los servicios que ofrece .NET Framework a las aplicaciones en ejecución son los siguientes”<sup>20</sup>:

- Administración de la memoria
- Sistema de tipos comunes
- Biblioteca de clases extensa
- Marcos y tecnologías de desarrollo
- Interoperabilidad de lenguajes
- Compatibilidad de versiones
- Ejecución en paralelo
- Compatibilidad con múltiples versiones (multi-targeting)

### **6.2.10.3 Visual Studio .NET**

Es un entorno gráfico de desarrollo integrado orientado a objetos (IDE) que permite a los desarrolladores crear, probar y depurar aplicaciones desarrolladas para la plataforma .NET, para todos los lenguajes .NET.

---

<sup>20</sup> MICROSOFT. Introducción a .NET Framework. Disponible en: <https://docs.microsoft.com/es-es/dotnet/framework/get-started/>



Este IDE de programación requiere el Framework .NET para poder ejecutarse, además de hacer mucho más ágil y práctico el desarrollo de aplicaciones. Este entorno de desarrollo incorpora los siguientes lenguajes de programación:

- Visual Basic.NET
- Visual C++.NET
- Visual J#.NET
- Visual C#.NET

#### **6.2.10.4 Características**

A continuación, se mencionan algunas de las características que han sido agregadas a la versión más reciente de Visual Studio 2017:

- Productividad mejorada
- Bases redefinidas
- Desarrollo en Azure simplificado
- Desarrollo de aplicaciones móviles de calidad

#### **6.2.10.5 Servicios Web (Microsoft .NET)**

Es un servicio web al que se puede acceder a través de internet mediante una URI (Uniform Resource Identification), estos servicios son utilizados por aplicaciones o páginas web, convirtiéndose en clientes de estos servicios web.

### 6.2.10.6 ADO .NET

Es un API que permite el acceso a fuentes de datos de .NET. heredando las características más relevantes de ADO, sin olvidar las propias como la posibilidad de trabajar de forma conectada como desconectada de las bases de datos.

Al trabajar de forma desconectada copia un conjunto de datos a un DataSet, en caso de que se presenten cambios en el DataSet estos cambios se efectúan sobre la fuente de datos.

#### 6.2.10.6.1 Principales características

Trabaja tanto de forma conectada como desconectado del origen de datos, permitiendo el acceso a la aplicación de más usuarios, tiene una fuerte integración con XML y ASP .NET, su uso es independiente del lenguaje de programación, y permite el acceso a bases de datos y a otras fuentes como lo son hojas de cálculo, XML, texto, etc.

#### 6.2.10.6.2 Las clases principales de ADO.NET:

- **Connection:** realiza la conexión con una fuente de datos.
- **Command:** permite hacer consultas u operaciones de modificación contra una fuente de datos.
- **DataReader:** permite acceder a los resultados de una consulta realizada contra una fuente de datos.
- **DataAdapter:** lee los datos y los carga en el dataset. Si ahí se producen cambios, el data adapter será el encargado de sincronizar todos los cambios en la fuente de datos.
- **DataSet:** permite almacenar y manipular datos en memoria.

### **6.2.11 BASES DE DATOS**

Se puede decir que a nivel informático, una base de datos es un archivo que tiene como principal funcionalidad el almacenar un conjunto de datos en memoria, los cuales se encuentran organizados mediante una estructura de datos. Las bases de datos se diseñan y crean para satisfacer las necesidades y requisitos en cuestión de información de uno o varios clientes en particular.

Dice la Dra. *María del Carmen Gómez Fuentes*, La mayoría de los sistemas computacionales utilizan una base de datos para manejar su información, por lo que es de suma importancia que los desarrolladores de software estén capacitados para su diseño, construcción y uso<sup>21</sup>.

“Una base de datos se puede percibir como un gran almacén de datos que se define y se crea una sola vez, y que se utiliza al mismo tiempo por distintos usuarios. En una base de datos todos los datos se integran con una mínima cantidad de duplicidad. De este modo, la base de datos no pertenece a un solo departamento, sino que se comparte por toda la organización. Además, la base de datos no sólo contiene los datos de la organización, también almacena una descripción de dichos datos. Esta descripción es lo que se denomina metadatos, se almacena en el diccionario de datos o catálogo y es lo que permite que exista independencia de datos lógica-física”<sup>22</sup>.

#### **6.2.11.1 Tipos de bases de datos**

Existen cuatro tipos diferentes de bases de datos:

- Bases de datos jerárquicas
- Bases de datos de red

---

21 Dra. María del Carmen Gómez Fuentes. Bases de datos

22 Mercedes Marques. Bases de datos. pág. 2

- Bases de datos relacionales
- Bases de datos orientadas a objetos

### **6.2.11.2 Sistemas de gestión de bases de datos**

Sistema de gestión de bases de datos o también conocido como (SGBD) por sus iniciales, son aplicaciones que permiten definir, crear y mantener las bases de datos, generalmente un SGBD proporciona los siguientes servicios:

- Definir una base de datos mediante un lenguaje de definición de datos.
- Insertar, eliminar actualizar, y consultar datos mediante el lenguaje de manejo de datos
- Acceso controlado a la base de datos

Mediante las siguientes características

- ✓ Sistema de seguridad
- ✓ Sistema de integridad y consistencia de los datos
- ✓ Sistema de control de concurrencia (acceso compartido a la BD)
- ✓ Sistema de control de recuperación
- ✓ Un diccionario de datos o catálogo.

Entre los SGBD más populares están:

- Microsoft SQL Server.
- IBMInformix.
- Oracle.
- MySQL.
- PostgreSQL.

- Microsoft Access

### 6.2.11.3 Ventajas de los sistemas de bases de datos

“Los sistemas de bases de datos presentan numerosas ventajas gracias y fundamentalmente, a la integración de datos y a la interfaz común que proporciona el SGBD. Estas ventajas se describen a continuación”<sup>23</sup>.

**Control sobre la redundancia de datos:** No se almacenan varias copias de los mismos datos. Sin embargo, en una base de datos no se puede eliminar la redundancia completamente, ya que en ocasiones es necesaria para modelar las relaciones entre los datos, o bien es necesaria para mejorar las prestaciones.

**Control sobre la consistencia de datos:** Eliminando o controlando las redundancias de datos se reduce en gran medida el riesgo de que haya inconsistencias. Si un dato está almacenado una sola vez, cualquier actualización se debe realizar sólo una vez, y está disponible para todos los usuarios inmediatamente. Si un dato está duplicado y el sistema conoce esta redundancia, el propio sistema puede encargarse de garantizar que todas las copias se mantengan consistentes.

**Compartición de datos:** En los sistemas de bases de Bases de datos, la base de datos pertenece a la empresa y puede ser compartida por todos los usuarios que estén autorizados. Además, las nuevas aplicaciones que se vayan creando pueden utilizar los datos de la base de datos existente

---

23 Mercedes Marques. Bases de datos. p. 9

**Mantenimiento de estándares:** Gracias a la integración es más fácil respetar los estándares necesarios, tanto los establecidos a nivel de la empresa como los nacionales e internacionales. Estos estándares pueden establecerse sobre el formato de los datos para facilitar su intercambio; pueden ser estándares de documentación, procedimientos de actualización y también reglas de acceso.

**Mejora en la integridad de datos:** La integridad de la base de datos se refiere a la validez de los datos almacenados. Normalmente, la integridad se expresa mediante restricciones o reglas que no se pueden violar. Estas restricciones se pueden aplicar tanto a los datos, como a sus relaciones, y es el SGBD quien se encargará de mantenerlas.

**Mejora en la seguridad:** La seguridad de la base de datos consiste en la protección de la base de datos frente a usuarios no autorizados. Sin unas buenas medidas de seguridad, la integración de datos en los sistemas de bases de datos hace que éstos sean más vulnerables que en los sistemas de ficheros. Sin embargo, los SGBD permiten mantener la seguridad mediante el establecimiento de claves para identificar al personal autorizado a utilizar la base de datos. Las autorizaciones se pueden realizar a nivel de operaciones, de modo que un usuario puede estar autorizado a consultar ciertos datos, pero no a actualizarlos

**Mejora en la accesibilidad a los datos:** Muchos SGBD proporcionan lenguajes de consulta o generadores de informes que permiten al usuario hacer cualquier tipo de consulta sobre los datos, sin que sea necesario que un programador escriba una aplicación que realice tal tarea

**Mejora en la productividad:** El SGBD proporciona muchas de las funciones estándar que el programador necesita escribir en un sistema de ficheros. A nivel básico, el SGBD proporciona todas las rutinas de manejo de ficheros típicas de los programas de aplicación. también proporcionan un entorno de cuarta generación consistente en un conjunto de herramientas que simplifican, en gran medida, el desarrollo de las aplicaciones que acceden a la base de datos. Gracias a estas herramientas, el programador puede ofrecer una mayor productividad en un tiempo menor.

**Mejora en el mantenimiento:** Los SGBD separan las descripciones de los datos de las aplicaciones. Esto es lo que se conoce como independencia de datos, gracias a la cual se simplifica el mantenimiento de las aplicaciones que acceden a la base de datos.

**Aumento de la concurrencia:** La mayoría de los SGBD gestionan el acceso concurrente a la base de datos y pueden garantizar que no ocurran problemas de interferencia entre los usuarios que acceden simultáneamente a la información, incluso previene la pérdida de la información.

**Mejora en los servicios de copias de seguridad y de recuperación ante fallos:** los SGBD actuales funcionan de modo que se minimiza la cantidad de trabajo perdido cuando se produce un fallo.

#### **6.2.11.4 Desventajas de los sistemas de bases de datos**

Como la mayoría de las cosas, los SGBD también tienen sus desventajas y algunas de estas se mencionan a continuación:

**Alta complejidad:** Los SGBD son conjuntos de programas muy complejos con una gran funcionalidad. Es preciso comprender muy bien esta funcionalidad para poder sacar un buen partido de ellos.

**Gran tamaño:** Los SGBD son programas complejos y muy extensos que requieren una gran cantidad de espacio en disco y de memoria para trabajar de forma eficiente.

**Coste económico:** El coste de un SGBD varía dependiendo del entorno y de la funcionalidad que ofrece. Además, hay que pagar una cuota anual de mantenimiento que suele ser un porcentaje del precio del SGBD. En los últimos años han surgido SGBD libres (open source) que ofrecen una gran funcionalidad y muy buenas prestaciones.

**Coste del equipamiento adicional:** Tanto el SGBD, como la propia base de datos, pueden hacer que sea necesario adquirir más espacio de almacenamiento. Además, para alcanzar las prestaciones deseadas, es posible que sea necesario adquirir una máquina más grande o una máquina que se dedique solamente al SGBD.

**Vulnerable a los fallos:** El hecho de que todo esté centralizado en el SGBD hace que el sistema sea más vulnerable ante los fallos que puedan producirse.

### 6.2.12 SQL

La sigla que se conoce como SQL corresponde a la expresión inglesa Structured Query Language (traducida al español como Lenguaje de Consulta Estructurado), es el lenguaje estándar de



definición, manipulación y control de bases de datos relacionales, en este lenguaje se debe indicar lo que se desea hacer es un lenguaje muy parecido al natural, para ser más preciso al inglés<sup>24</sup>.

### **6.2.13 SQL SERVER**

Es un sistema de administración y análisis de bases de datos relacionales de Microsoft, tiene características como control de transacciones, excepción y manejo de errores, declaración de variable, entre otras

“SQL Server 2014 también proporciona nuevas soluciones de copia de seguridad y de recuperación ante desastres, así como de arquitectura híbrida con Windows Azure, lo que permite a los clientes utilizar sus actuales conocimientos con características locales que aprovechan los centros de datos globales de Microsoft. Además, SQL Server 2014 aprovecha las nuevas capacidades de Windows Server 2012 y Windows Server 2012 R2 para ofrecer una escalabilidad sin parangón a las aplicaciones de base de datos en un entorno físico o virtual”<sup>25</sup>.

### **6.2.14 ASP .NET**

Es la plataforma de Microsoft que permite desarrollar aplicaciones web, las cuales superan las paginas HTML tradicionales, es una solución poderosa y fácil de utilizar, diseñada para el desarrollo de aplicaciones web. Mediante el uso de esta tecnología se pueden crear paginas complejas permitiendo una comunicación mucho más agradable con el cliente y no como las tradicionales páginas web que solo eran un despliegue de información.

---

24 MARTIN ESCOFET, Carmen. El lenguaje SQL. p. 5

25 MICROSOFT SQL Server. 2014. Disponible en: <https://msdn.microsoft.com/es-es/library/bb545450.aspx>

Permite el uso de lenguajes de programación como C#, Visual Basic, J# entre otros, en la mayoría de los casos, los usuarios utilizan como entorno de desarrollo el Visual Studio .NET ya que posee una serie de características que facilitan las tareas del desarrollador como lo son el desarrollo de aplicaciones mediante el uso de componente visuales<sup>26</sup>.

---

<sup>26</sup> ARIAS Ángel. Aprende a programar ASP .NET y C# 2ª ed.

## **7 CAPÍTULO IV - INGENIERÍA DEL PROYECTO**

### **7.1 DESARROLLO DEL PROYECTO**

Continuando con el desarrollo del proyecto, se procede con el análisis, diseño e implementación del aplicativo, el cual supla las necesidades con las que cuenta la Institución Educativa Nueva Visión de Honduras en cuestiones tecnológicas, para el registro control y consulta de notas.

Para el óptimo desarrollo de este aplicativo se cuenta como guía de trabajo la metodología RUP, la cual guía paso a paso en cada una de las fases que se deben implementar hasta conseguir el producto final.

De acuerdo a esta metodología y teniendo en cuenta las fases que esta propone, para el buen cumplimiento de este proyecto, se procede con el desarrollo de la aplicación mediante el uso e implementación de las siguientes fases:

#### **7.1.1 FASE DE INICIO**

Durante esta fase se procede con la exploración y análisis de requerimientos con los que contará la aplicación.

##### **7.1.1.1 Recolección de información**

Mediante el uso de la observación, que es un instrumento de recolección de información se vio que la Institución Educativa Nueva Visión de Honduras, no contaba con un software que realice el proceso de registro, control y consulta de notas; debido a esto se le propuso al rector de esta institución educativa el análisis, diseño y creación de un software que realice esta labor. Con el objetivo de cumplir con lo planteado se llevaron a cabo una serie de reuniones con el docente encargado de asistir durante el desarrollo del proyecto, recopilando la información necesaria para el desarrollo del software, obteniendo los siguientes documentos.

### 7.1.1.2 Documentos de procesos proporcionados por la Institución

815 - INST EDUC. NUEVA VISION DE HONDURAS - PRINCIPAL  
LISTA AUXILIAR PARA FIRMA DE RECIBIDO  
AÑO LECTIVO 2018

02-09-18  
Grupo: M - 7.01  
PERIODO: 3

No.	ESTUDIANTE	CODIGO	RECIBI BOLETIN
1	AGRONO TAMAYO EMELY SOFIA	98	
2	ARARAT MONTAÑO WILMER GABRIEL	99	
3	BALANTA CAICEDO WENDY NAYELY	100	
4	CAICEDO CAICEDO SANTIAGO	101	
21	CAICEDO MINA YAJARA	102	
5	CAICEDO SANDOVAL JUAN DAVID	103	RETRADO
6	CAICEDO TRUJILLO GEISER DAVID	104	
7	CAICEDO TRUJILLO YURY ALEXANDRA	105	
8	CAICEDO VASQUEZ INGRID NATALIA	106	
9	CANO MONTAÑO YENCY LORENA	107	
10	CARABALI CARABALI YNGRI LICET	108	
11	CASTILLO BALANTA LUISA MANUELA	109	
12	CHOCUE SANDOVAL LAURA DANIELA	110	
13	MARROQUIN MARROQUIN LESLY FACLA	111	
14	NAZARIT CAICEDO VIVIAN ALEXANDRA	112	
15	NAZARIT MINA JUAN PABLO	113	
16	RODALLEGA CAICEDO SARA LIDIA	114	
17	SOLIS CAICEDO MARLON STIVEN	115	
18	SOLIS CAICEDO SARAY	116	
19	SOLIS CAICEDO SHARY YULIANA	117	
20	TRUJILLO SANDOVAL DUVAN DARIO	118	

Fecha: \_\_\_\_\_


PROCESADO E IMPRESO POR 

Ilustración 3 Lista para firma de recibido

PLANILLA DE REGISTRO Y SEGUIMIENTO ACADÉMICO		INSTITUCIÓN EDUCATIVA		ANNO LECTIVO	PERIODO	GRADO	PARALELO							
		#16 - INST. EDUC. NUEVA VISION DE HONDURAS		2014	M	5.º	4							
DOCENTE		1-- SALINAS CARABALJ OFELIA		ASIGNATURA J01 - DIMENSION COMUNICATIVA										
NOMBRE DEL ESTUDIANTE	Nº	DESEMPEÑOS PERIODOS ANTERIORES			USQUE EN ESTAS COLUMNAS LOS CÓDIGOS CORRESPONDIENTES A ESTA MATERIA Y GRADO SEGUN LA CARTILLA DE LOGROS					Nº	CÓDIGO DE CALIFICACION	MONEDAS	VALORES	
		I	II	III										
APONZA DIVAN FELIPE	1	3.0	3.0	3.0	80	101					1	35		
ARARAT ARBOLEDA JOSE	2	3.0	3.0	3.0	80	101					2	30		
ARARAT ARBOLEDA SAMUEL	3	3.0	3.0	3.0	80	101					3	30		
ARBOLEDA NAZARIT BRAYAN ESTIVEN	4	3.0	3.5	3.0	80	101					4	30		
BALANTA SANDOVAL SHAROL SOFIA	5	4.0	3.5	3.5	80	101					5	34		
BERMUCHEZ MINA DANIEL	6	3.0	3.0	3.5	80	101					6	30		
CAICEDO CAICEDO DILAN DAVID	7	3.5	3.0	3.5	80	101					7	35		
CAICEDO MEZU VALENTINA	8	4.0	3.7	3.9	79	100					8	41		
CAICEDO SALINAS ALEX	9	3.0	3.0	3.0	80	101					9	30		
CARABALI BALANTA DERIAN DAVID	10	4.0	4.0	4.0	79	100					10	45		
CARABALI LEON ABIGAIL	11	3.7	3.7	3.9	79	100					11	45		
CARABALI NAZARIT SEBASTIAN	12	3.0	3.0	3.0	80	101					12	30		
CENCIO POPO JOSE MIGUEL	13	3.0	3.0	3.0	79	100					13	41		
CHARRUPI TRUJILLO CARLOS JHOSUAC	14	4.0	3.8	3.7	80	101					14	36		
CHOCO MINA SARI SOFIA	15	3.0	3.5	3.8	80	101					15	35		
DIAZ SALINAS LUIS ANDRES	17	4.0	4.0	4.0	79	100					17	43		
MINA CARABALI ESTEBAN	18	3.5	3.0	3.0	80	101					18	35		
MONTAÑO RODALLEGA ISABEL CRISTINA	19	3.5	3.7	3.0	80	101					19	30		
MONTAÑO RODALLEGA MARIA ISABEL	20	3.5	3.7	3.9	80	101					20	36		
RAMOS AMBUILA ANGEL DAVID	21	3.2	3.0	3.0	80	101					21	35		
SANDOVAL ARARAT CARLOS ANDRES	23	4.0	3.5	3.9	80	101					23	39		
SOLIS APONZA DERIAN ESTIVEN	24	3.0	3.5	3.1	80	101					24	35		
SOLIS CARABALI JUAN CARLOS	25	3.0	3.0	3.0	80	101					25	30		
SOLIS VERGARA VALERIA	27	3.7	4.0	3.9	80	101					27	35		

Ofelia Salinas

Carabali modo

logros

monedas

Nota

Ilustración 4 Planilla de registro y seguimiento académico

INST. EDUC. NUEVA VISION DE HONDURAS -- PRINCIPAL  
AÑO LECTIVO: 2018/2018  
LISTADO DE PUESTOS POR PROMEDIO

02-09-18  
PERIODO: 3

GRUPO: M - 7 . 01

	Puesto	Prom.
13	MARROQUIN MARROQUIN LESLY PAOLA	1 4.12
15	NAZARIT MINA JUAN PABLO	2 4.08
20	TRUJILLO SANDOVAL DUVAN DARIO	3 3.83
16	RODALLEGA CAICEDO SARA LIDIA	4 3.79
1	AGRONO TAMAYO EMELY SOFIA	5 3.78
6	CAICEDO TRUJILLO GEISER DAVID	6 3.69
12	CHOCUE SANDOVAL LAURA DANIELA	7 3.63
14	NAZARIT CAICEDO VIVIAN ALEXANDRA	8 3.58
17	SOLIS CAICEDO MARLON STIVEN	8 3.58
19	SOLIS CAICEDO SHARY YULIANA	9 3.56
2	ARARAT MONTAÑO WILMER GABRIEL	10 3.37
9	CANO MONTAÑO YENCY LORENA	11 3.33
18	SOLIS CAICEDO SARAY	12 3.31
4	CAICEDO CAICEDO SANTIAGO	13 3.23
8	CAICEDO VASQUEZ INGRID NATALIA	14 3.19
3	BALANTA CAICEDO WENDY NAYELY	15 3.13
7	CAICEDO TRUJILLO YURY ALEXANDRA	15 3.13
11	CASTILLO BALANTA LUISA MANUELA	16 3.01
10	CARABALI CARABALI YNGRI LICET	17 2.81
21	CAICEDO MINA YAJAIRA	18 2.40

Ilustración 5 *Listado de puestos por promedio*

815 INST. EDUC. NUEVA VISION DE HONDURAS ---PRINCIPAL  
 LISTADO DE ALUMNOS CON AREAS PERDIDAS, AÑO LECTIVO 2018/2018

GRADO: 7 - 01

Periodo 3

No	ALUMNO	MATERIAS PERDIDAS
2	ARARAT MONTAÑO WILMER GABRIEL	2.8 MATEMATICAS
6	CAICEDO TRUJILLO GEISER DAVID	2.5 MATEMATICAS
7	CAICEDO TRUJILLO YURY ALEXANDRA	2.5 INGLES 2.5 MATEMATICAS
8	CAICEDO VASQUEZ INGRID NATALIA	2.5 CIENCIAS NATURALES Y EDUCACION 2.0 INGLES 2.5 MATEMATICAS
9	CANO MONTAÑO YENCY LORENA	2.0 EDUCACION ARTISTICA 2.5 MATEMATICAS
10	CARABALI CARABALI YNGRI LICET	2.8 CIENCIAS NATURALES Y EDUCACION 2.0 INGLES 2.0 MATEMATICAS
11	CASTILLO BALANTA LUISA MANUELA	2.7 EDU. ETICA Y VALORES HUMANOS 2.0 INGLES
12	CHOCUE SANDOVAL LAURA DANIELA	2.6 EDU. ETICA Y VALORES HUMANOS
16	RODALLEGA CAICEDO SARA LIDIA	2.8 MATEMATICAS
17	SOLIS CAICEDO MARLON ESTIVEN	2.8 MATEMATICAS
18	SOLIS CAICEDO SARAY	2.5 INGLES
19	SOLIS CAICEDO SHARY YULIANA	2.8 INGLES
21	CAICEDO MINA YAJAIRA	2.5 CIENCIAS SOCIALES 2.0 EDUCACION ARTISTICA 2.0 INGLES 2.5 MATEMATICAS

Ilustración 6 Listado de alumnos con asignaturas perdidas

# INSTITUCIÓN EDUCATIVA NUEVA VISIÓN DE HONDURAS

Resolución de Aprobación No. 0449 del 25 de Abril de 2004.

Buenos Aires - Cauca



## SISTEMA INSTITUCIONAL DE EVALUACIÓN DE LOS ESTUDIANTES - (DECRETO 1290 DE 2009)

### INFORME DESCRIPTIVO VALORATIVO 2018

#### TERCER PERIODO

Grado: 7.01 No. 1

Estudiante: **AGRONO TAMAYO EMELY SOFIA**

CODIGO	AREA y/o ASIGNATURA	INTENSIDAD HORARIA	FALTAS ASIST.	CONSOLIDADO			DESEMPEÑO PERIODO
				I	E	IV	
A01	CIENCIAS NATURALES Y EDUCACION	4		3.9	4.2	3.7	BÁSICO
B01	CIENCIAS SOCIALES	4		3.5	4.1	4.2	ALTO
B02	CATEDRA DE LA PAZ	1		3.0	4.0	4.0	ALTO
C01	EDUCACION ARTISTICA	2		4.0	4.0	4.5	ALTO
D01	EDU.ETICA Y VALORES HUMANOS	1		5.0	4.8	3.4	BÁSICO
ED1	EDUCACION FISICA	2		3.4	3.7	3.5	BÁSICO
G01	LENGUA CASTELLANA	5		4.2	3.8	3.8	BÁSICO
G02	INGLES	2		3.5	3.7	3.7	BÁSICO
H01	MATEMATICAS	5	1	3.5	4.0	3.5	BÁSICO
IB1	TECNOLOGIA E INFORMATICA	2		4.0	4.2	3.5	BÁSICO
Z01	CONVIVENCIA EN COMUNIDAD	0		4.5	5.0	4.2	ALTO
DESEMPEÑO DEL ESTUDIANTE				3.90	4.04	3.76	BÁSICO
PUESTO QUE OCUPA EN EL SALÓN:				3	2	5	

#### DEFINICIÓN DE TÉRMINOS

SUPERIOR [4.8 - 5.0]	Estudiante que ALCANZA todos los desempeños propuestos del periodo y desarrolla actividades curriculares que exceden las exigencias esperadas.
ALTO [4.0 - 4.5]	Estudiante que ALCANZA todos los desempeños propuestos del periodo y desarrolla actividades curriculares básicas.
BÁSICO [3.0 - 3.9]	Estudiante que ALCANZA los desempeños mínimos y desarrolla un mínimo de actividades curriculares requeridas.
BAJO [1.0 - 2.9]	Estudiante que NO ALCANZA los desempeños mínimos y requiere actividades de refuerzo y superación. No desarrolla el mínimo de actividades curriculares requeridas.

#### CIENCIAS NATURALES Y EDUCACION

DOC.: RUTH IVONNE MORENO

##### FORTALEZAS

MANIFIESTA INTERES POR SU PROCESO DE APRENDIZAJE

#### CIENCIAS SOCIALES

DOC.: JESUS ANTONIO BALANTA

##### FORTALEZAS

ESTABLECE RELACIONES ENTRE LAS CARACTERÍSTICAS DE LAS REGIONES DE AMÉRICA

REFLEXIONA SOBRE LA IMPORTANCIA DEL RESPETO Y VALORACIÓN DE LOS DERECHOS HUMANOS EN LA ACTUALIDAD

#### CATEDRA DE LA PAZ

DOC.: FRANCA MILEYDI ALARCON

##### DIFICULTADES

CON DIFICULTAD IDENTIFICA EL PROCESO HISTÓRICO A TRAVÉS DEL CUAL SE DESARROLLA LA GLOBALIZACIÓN COMO SISTEMA ECONÓMICO, SOCIAL, CULTURAL Y POLÍTICO IMPERANTE EN EL CONTEXTO ACTUAL.

#### EDUCACION ARTISTICA

DOC.: FRANCA MILEYDI ALARCON

##### FORTALEZAS

REALIZA TRABAJOS MANUALES CON DESTREZA Y HABILIDAD.

#### EDU.ETICA Y VALORES HUMANOS

DOC.: HECTOR TORRES

##### FORTALEZAS

EL ESTUDIANTE ALCANZO SATISFACTORIAMENTE LOS LOGROS DEL PROYECTO DE AULA REALIZADOS EN EL CURSO

#### EDUCACION FISICA

DOC.: MARIA VICTORIA LARRAHONDO

##### DIFICULTADES

NO PROMUEVE LA CREACION DE GRUPOS JUVENILES PARA EL USO CREATIVO DEL TIEMPO LIBRE, COMO ESTRATEGIA DE PREVENCIÓN DEL CONSUMO DE SUSTANCIAS PSICOACTIVAS.

CON DIFICULTAD ORGANIZA Y LIDERA ACTIVIDADES DIRIGIDAS A DINAMIZAR, RECUPERAR Y CONSERVAR EL MEDIO AMBIENTE.

#### LENGUA CASTELLANA

DOC.: ESPERANZA CARABALI

##### DIFICULTADES

CON DIFICULTAD DEFINE UNA TEMATICA PARA LA ELABORACION DE UN TEXTO ORAL CON FINES ARGUMENTATIVOS.

#### INGLES

DOC.: GINA MARCELA RODRIGUEZ

##### FORTALEZAS

Ilustración 7 Informe de periodo



### 7.1.1.3 Requerimientos

El principal objetivo de la ingeniería de requerimientos es: recopilar, analizar y verificar las necesidades del cliente, tiene como prioridad especificar de manera correcta los requisitos y mejorar la forma de comprender el sistema que se desea lograr.

“La meta de la ingeniería de requerimientos es entregar una especificación de requerimientos de software correcta y completa. La ingeniería de requerimientos apunta a mejorar la forma en que comprendemos y definimos sistemas de software complejos”<sup>27</sup>. Según la IEEE los requerimientos son Condiciones o capacidades requeridas por el usuario para resolver un problema o alcanzar un objetivo. De acuerdo a esto, y a la documentación presentada por la institución el sistema debe cumplir con los siguientes requerimientos:

#### 7.1.1.3.1 Requerimientos funcionales

**Tabla 1 RF Iniciar sesión**

Iniciar sesión	
Nº	Descripción
	El sistema debe validar el acceso, solo permite el ingreso de usuarios registrados en la base de datos de la aplicación (Administrador, Docentes, Estudiantes)
	El sistema debe restringir el acceso que tienen los actores dentro de la aplicación, de acuerdo al tipo de usuario que ingresa, habilita la sesión correspondiente al usuario registrado. los cuales pueden ser:

---

27 GIL, Gustavo Daniel. herramienta para implementar LEL y escenarios (TILS)

	Administrador Docente Estudiante
	Validar que la información ingresada (usuario y contraseña) sean correctos. Si la información es correcta permita ingresar al sistema de lo contrario solicite nuevamente nombre de usuario y contraseña

RF Iniciar sesión – fuente propia

**Tabla 2 RF Administrador**

Administrador	
<b>Nº</b>	<b>Descripción</b>
	El sistema permitirá administrar (Crear, eliminar, actualizar y consultar) los datos de los docentes registrados en él
	El sistema permite establecer las asignaturas que dicta un docente en los distintos grados
	El sistema permitirá administrar (Crear, eliminar, actualizar y buscar) los datos de los estudiantes en el sistema
	El sistema permitirá administrar (Crear, eliminar, actualizar y buscar) los datos de las asignaturas en el sistema
	El sistema permitirá administrar (Crear, eliminar, actualizar y buscar) los datos de los grados en el sistema
	El sistema genera los reportes correspondientes

Requerimientos funcionales del administrador – fuente propia

**Tabla 3 RF Docentes**

Docentes	
Nº	Descripción
	Iniciar sesión
	Registrar las notas de los estudiantes, en los grados en los cuales dicta clases
	Poder visualizar las notas ingresadas a los estudiantes a su cargo

Requerimientos funcionales del docente – fuente propia

**Tabla 4 RF Estudiantes**

Estudiantes	
Nº	Descripción
	Visualizar el promedio de notas en cada una de las distintas asignaturas
	Visualizar las notas que tiene registradas en las distintas asignaturas

Requerimientos funcionales del estudiante – fuente propia

#### 7.1.1.3.2 Requerimientos no funcionales

**Tabla 5 RNF Sistema**

Sistema	
Nº	Descripción
	El sistema permita una fácil usabilidad al usuario final
	El sistema será desarrollado mediante Microsoft Visual Studio Express ASP .NET (C#)
	El motor de la base de datos a utilizar será SQL Server 2017 Express

	El sistema estará disponible vía Intranet mediante el uso de un navegador web
	El sistema tendrá su propio manual de usuario para su fácil entendimiento

Requerimientos no funcionales del sistema – fuente propia

### 7.1.2 FASE DE ELABORACIÓN

Es el proceso durante el cual se planifica una solución de software a un problema planteado en donde los desarrolladores manejan la complejidad que proporcionan los programas informáticos, además de disminuir en gran medida los riesgos que estos proponen.

Según Pressman, el diseño del software es realmente un proceso de muchos pasos pero que se clasifican dentro de uno mismo. En general, la actividad del diseño se refiere al establecimiento de las estructuras de datos, la arquitectura general del software, representaciones de interfaz y algoritmos. El proceso de diseño traduce requisitos en una representación de software [PRR98]<sup>28</sup>.

Se deben conocer técnicas y metodologías en la ingeniería de software para tener un buen entendimiento en el comportamiento de los sistemas con los cuales se enfrentan los desarrolladores de software, lograr entender lo que el cliente desea transmitir, entender a cabalidad los requerimientos para con el software deseado. Además, es de vital importancia tener muy claros los procesos con los cuales contará el sistema y las entidades que hay intervienen:

- Dividir el sistema y ver cómo interactúan sus partes

---

28 CAPÍTULO 2: Ingeniería de Software, Análisis y Diseño. Disponible en:  
[http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/fuentes\\_k\\_jf/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf)

- Entre más claro el modelado del software es mucho más fácil su desarrollo
- Entre mejor se encuentre el modelado más fácil se cumplen las exigencias del cliente

Es debido a esto, y teniendo en cuenta los requerimientos funcionales de la aplicación se procede con la especificación de los casos de uso con los cuales contará la aplicación, logrando de esta manera un fácil desarrollo de la misma.

### 7.1.2.1 Diagrama de casos de uso

#### Diagrama general de casos de uso

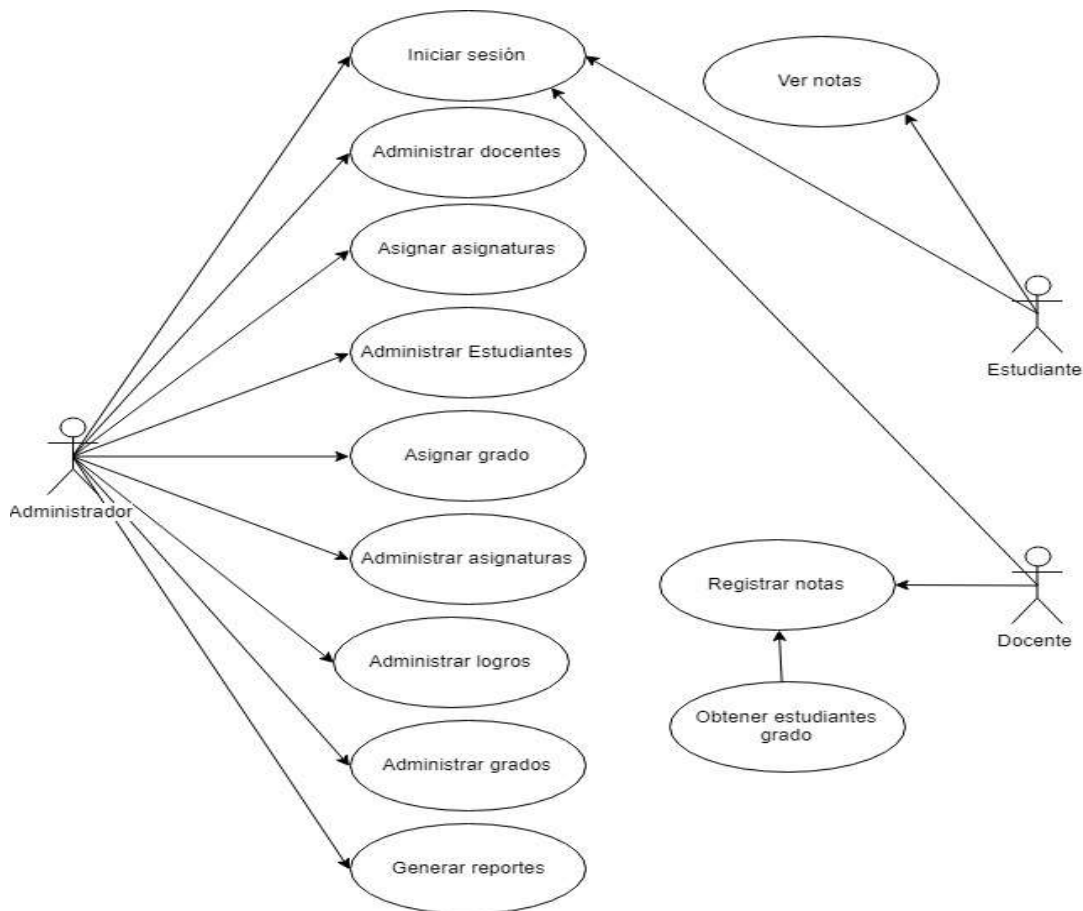


Ilustración 8 Diagrama general de CU– fuente propia

## Iniciar sesión

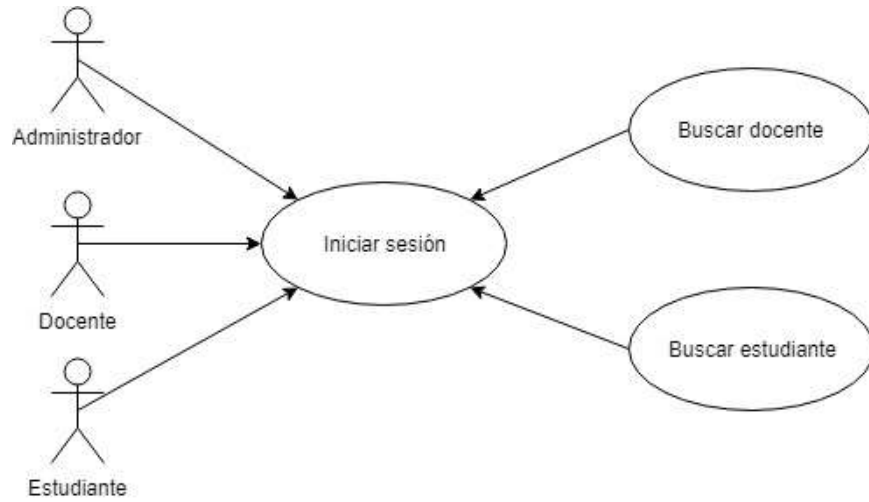


Ilustración 9 Diagrama de CU Iniciar sesión– fuente propia

## Administrar docentes

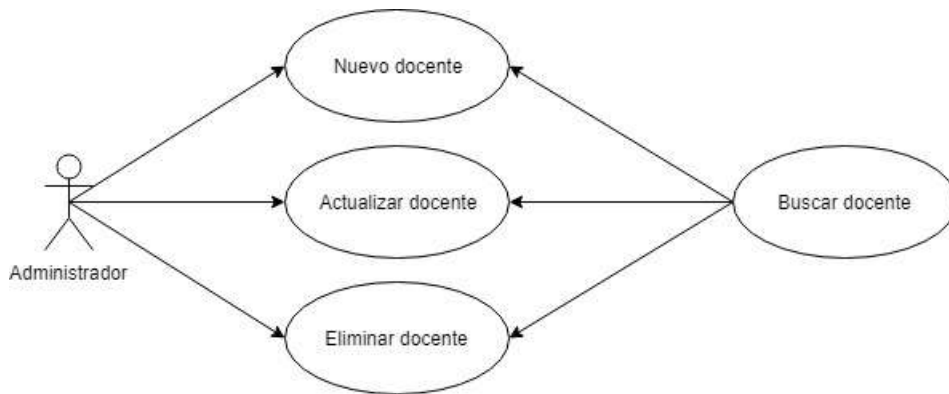


Ilustración 10 Diagrama de CU Administrar docentes– fuente propia

## Administrar estudiantes

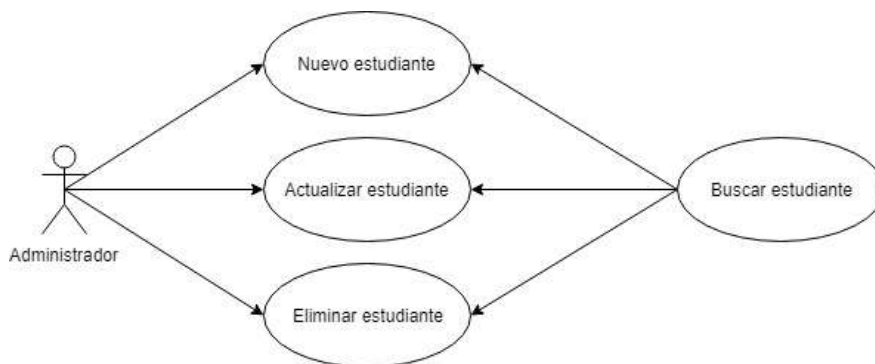


Ilustración 11 Diagrama de CU Administrar estudiantes– fuente propia

## Administrar asignaturas

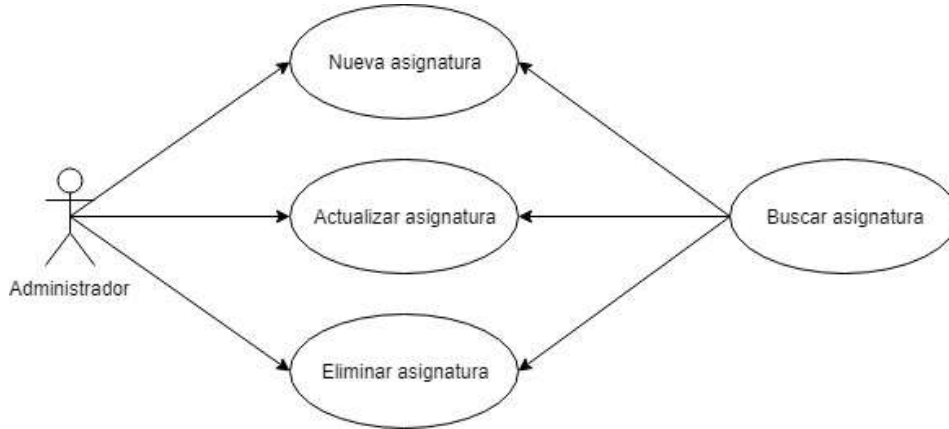


Ilustración 12 Diagrama de CU Administrar asignaturas– fuente propia

## Administrar grados

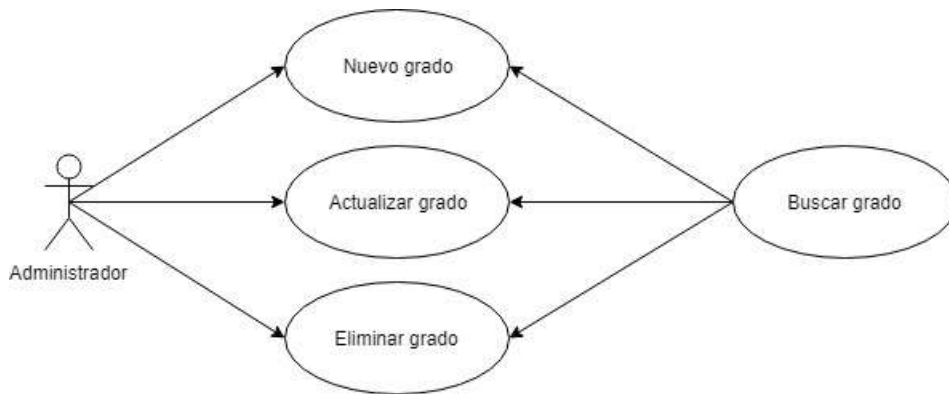


Ilustración 13 Diagrama de CU Administrar grados– fuente propia

## Administrar logros

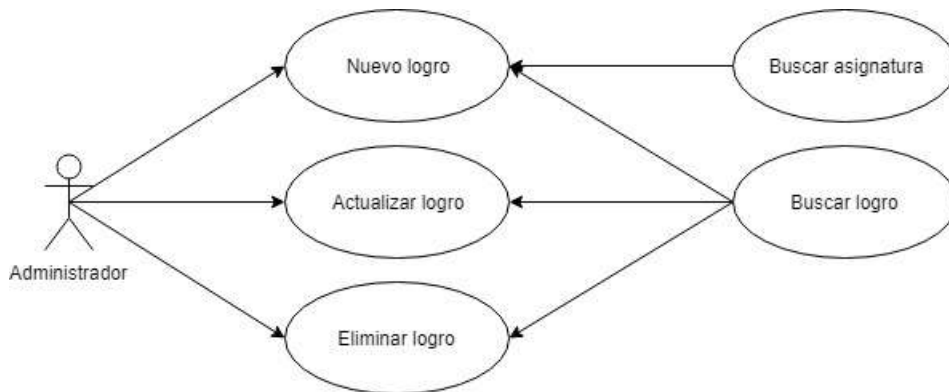


Ilustración 14 Diagrama de CU Administrar logros– fuente propia

## Asignar asignaturas

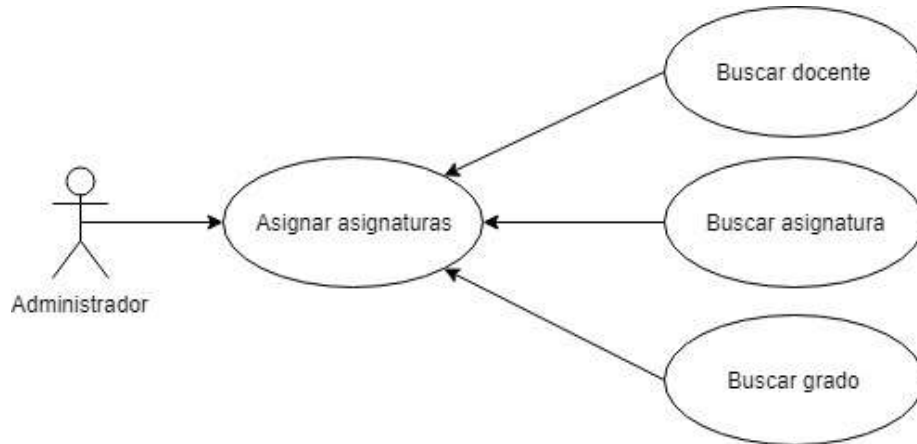


Ilustración 15 Diagrama de CU Asignar asignaturas–fuente propia

## Asignar grado

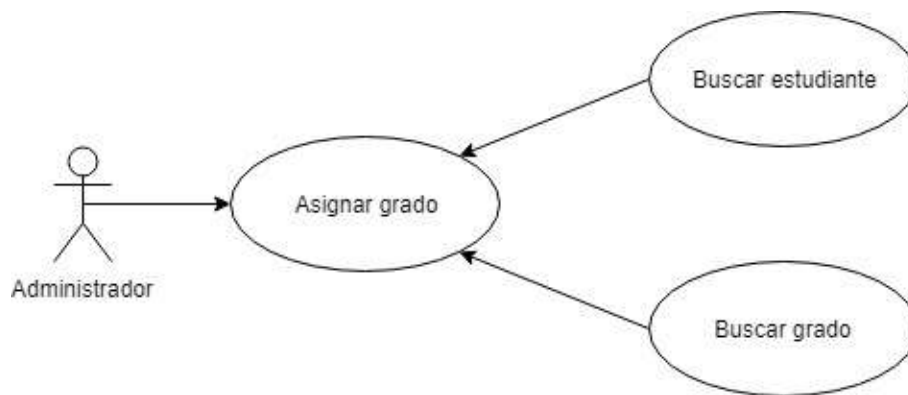


Ilustración 16 Diagrama de CU Asignar grado–fuente propia

## Administrar notas

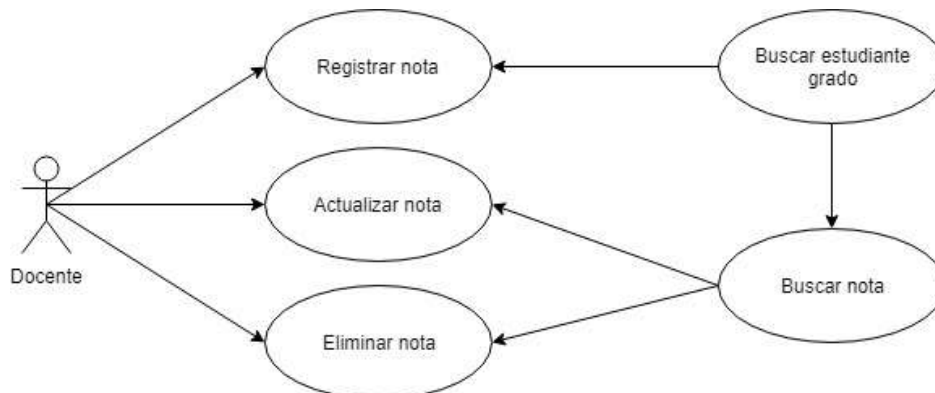


Ilustración 17 Diagrama CU Administrar nota–fuente propia



## Ver notas

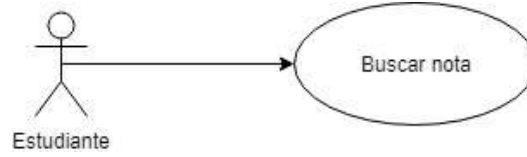


Ilustración 18 Diagrama CU Buscar nota– fuente propia

## Generar reporte

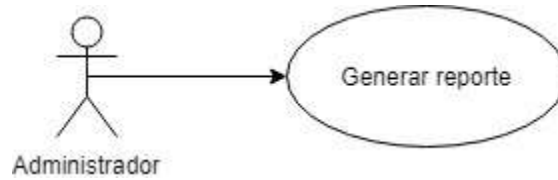


Ilustración 19 Diagrama de CU Generar reporte– fuente propia

### 7.1.2.2 Casos de uso

Tabla 6 CU Iniciar sesión

Aplicación	
Iniciar sesión	
Descripción del caso de uso	
Nombre:	Iniciar sesión
Actores	Administrador Docente Estudiante
Función:	Validar el acceso al sistema
Descripción:	Esta funcionalidad permite el ingreso al sistema de los usuarios registrados
Precondición:	El usuario se encuentre registrado en la base de datos del sistema

Éxito	Usuario y contraseñas son correctos. Ingresa al sistema
Fracaso	Usuario o contraseña son incorrectos. Usuario no puede ingresar al sistema
Iniciador	El actor abre la página principal de la aplicación
Escenario principal	<ol style="list-style-type: none"> <li>1. El actor diligencia los datos solicitados por el sistema (Usuario, contraseña)</li> <li>2. El sistema busca en la base de datos, el usuario que intenta ingresar al sistema, valida que coincidan el usuario y la contraseña, si estos son correctos permite el acceso, de lo contrario solicita datos válidos.</li> </ol>

Iniciar sesión – fuente propia

**Tabla 7 CU Nuevo docente**

Aplicación	
Administrar docentes	
Descripción del caso de uso	
Nombre:	Nuevo docente
Actores	Administrador
Función:	Registrar un docente en el sistema
Descripción:	El administrador podrá ingresar un docente en la base de datos, diligenciando los datos solicitados por el sistema.
Precondición:	Haber iniciado sesión como administrador
Éxito	Se guarda la información del docente en el sistema
Fracaso	No se guarda la información del docente en el sistema

Iniciador	El administrador da clic sobre la opción docente
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Docente</li> <li>2. El sistema abre el formulario (Administrar docentes)</li> <li>3. El administrador da clic sobre la opción Nuevo</li> <li>4. El sistema abre un nuevo formulario solicitando los datos del docente</li> <li>5. El administrador ingresa los datos solicitados por el sistema y da clic sobre la opción Guardar</li> <li>6. El sistema valida los datos ingresados (campos obligatorios y duplicidad de identificación). Si los datos son válidos guarda la información de lo contrario solicita que se ingresen datos válidos.</li> <li>7. El sistema muestra un mensaje informado que se ingresó un nuevo docente</li> </ol>

Nuevo docente – fuente propia

**Tabla 8 CU Eliminar docente**

Aplicación	
Administrar docentes	
Descripción del caso de uso	
Nombre:	Eliminar docente
Actores	Administrador
Función:	Eliminar un docente del sistema

Descripción:	El administrador puede eliminar un docente de la base de datos del sistema
Precondición:	Que se encuentre en la base de datos de la aplicación, el docente a eliminar Saber el número de cedula del docente a eliminar El docente a eliminar no puede tener asignaturas asignadas
Éxito	Se elimina el docente de la base de datos del sistema
Fracaso	No se elimina el docente la base de datos del sistema
Iniciador	El administrador da clic sobre la opción Docente, Eliminar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Docente</li> <li>2. El sistema abre el formulario Administrar docentes</li> <li>3. El administrador da clic sobre la opción Eliminar</li> <li>4. El sistema abre un formulario para eliminar un docente, solicitando el número de cédula del docente a eliminar</li> <li>5. El administrador digita el número de cedula del docente a eliminar y da clic sobre la opción eliminar</li> <li>6. El sistema busca al docente mediante el número de cedula diligenciado, si este se encuentra y no tiene asignaturas asignadas, lo elimina</li> <li>7. El sistema muestra un mensaje informando que se eliminó el docente</li> </ol>

Eliminar docente – fuente propia

**Tabla 9 CU Actualizar docente**

Aplicación	
Administrar docentes	
Descripción del caso de uso	
Nombre:	Actualizar docente
Actores	Administrador
Función:	Actualiza los datos de un docente
Descripción:	El administrador podrá actualizar uno o varios datos de un docente registrado en la base de datos del sistema
Precondición:	Que se encuentre en la base de datos de la aplicación, el docente al cual se le desean actualizar los datos  Buscar al docente al cual se le desean actualizar los datos
Éxito	Se guardan los cambios realizados sobre el docente que se desea actualizar
Fracaso	No se guardan los cambios realizados sobre el docente que se desea actualizar
Iniciador	El administrador da clic sobre la opción Docente, Actualizar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Docente</li> <li>2. El sistema abre el formulario Administrar docentes</li> <li>3. El administrador da clic sobre la opción Actualizar</li> <li>4. El sistema habilita el formulario para actualizar los datos de un docente</li> </ol>

	<p>5. El administrador busca y selecciona el docente que desea actualizar</p> <p>6. El sistema habilita el formulario con los datos del docente a actualizar</p> <p>7. El administrador modifica uno o varios datos del docente que desea actualizar y da clic sobre la opción Guardar</p> <p>8. El sistema valida los nuevos datos ingresados (campos obligatorios y duplicidad en la identificación del docente). Si estos datos son válidos actualiza los datos del docente, de lo contrario solicita que se ingresen datos válidos.</p> <p>9. El sistema muestra un mensaje informando que se actualizaron los datos de un docente</p>
--	--

Actualizar docente – fuente propia

**Tabla 10 CU Buscar docente**

Aplicación	
Administrar docentes	
Descripción del caso de uso	
Nombre:	Buscar docente
Actores	Administrador
Función:	Buscar los docentes en la aplicación
Descripción:	Buscar los docentes que se encuentran registrados en la base de datos de la aplicación

Precondición:	El docente a buscar se encuentre registrado en la base de datos de la aplicación
Éxito	Que se encuentren los datos del docente buscado
Fracaso	Que no se encuentren los datos del docente buscado
Iniciador	Diligenciar los datos del docente a buscar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Docente</li> <li>2. El sistema abre el formulario Administrar docentes</li> <li>3. El administrador da clic sobre la opción Buscar</li> <li>4. El sistema habilita el formulario para buscar un docente</li> <li>5. El administrador diligencia el número de cedula del docente a buscar y da clic sobre la opción buscar</li> <li>6. El sistema muestra el docente que coincide con la búsqueda realizada</li> </ol>

Buscar docente – fuente propia

**Tabla 11 CU Asignar asignaturas**

Aplicación	
Administrar docentes	
Descripción del caso de uso	
Nombre:	Asignar asignaturas
Actores	Administrador
Función:	Asignar carga al docente
Descripción:	Asignarle a un docente las asignaturas que dicta en los diferentes grados

Precondición:	<p>Que el docente se encuentre registrado en la base de datos del sistema</p> <p>Que las asignaturas se encuentren registradas en la base de datos del sistema</p> <p>Que se encuentre creado el grado al que se asigna</p>
Éxito	Que se le asigne asignatura y grado al docente seleccionado
Fracaso	Que no se le asigne asignatura y grado al docente seleccionado
Iniciador	<p>Buscar y seleccionar a docente y dar clic sobre la opción Asignaturas</p>
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador busca y selecciona un docente y da clic sobre la opción Asignaturas</li> <li>2. El sistema habilita el formulario para asignar asignaturas</li> <li>3. El sistema solicita los campos necesarios para asignar asignaturas a un docente</li> <li>4. El administrador selecciona el Grado, Grupo y la asignatura, y da clic sobre la opción Asignar</li> <li>5. El sistema valida que la asignatura no se encuentre asignada en el grado y grupo seleccionado, si es así la asigna de lo contrario informa que ya se ha asignado</li> <li>6. El sistema muestra un mensaje informando que se asignó la asignatura</li> </ol>

Asignar docente, asignatura, grado – fuente propia



**Tabla 12 CU Nuevo estudiante**

Aplicación	
Administrar estudiantes	
Descripción del caso de uso	
Nombre:	Nuevo estudiante
Actores	Administrador
Función:	Registrar un estudiante en el sistema
Descripción:	El administrador podrá ingresar un estudiante en la base de datos del sistema, diligenciando los datos solicitados.
Precondición:	Haber iniciado sesión como administrador
Éxito	Se guarda la información del estudiante en el sistema
Fracaso	No se guarda la información del estudiante en el sistema
Iniciador	El administrador da clic sobre la opción Estudiantes
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Estudiantes</li> <li>2. El sistema abre el formulario (Administrar estudiantes)</li> <li>3. El administrador da clic sobre la opción Nuevo</li> <li>4. El sistema abre un nuevo formulario solicitando los datos del estudiante</li> <li>5. El administrador ingresa los datos solicitados por el sistema y da clic sobre la opción Guardar</li> <li>6. El sistema valida los datos ingresados (campos obligatorios y duplicidad de identificación). Si los datos</li> </ol>

	<p>son válidos guarda la información de lo contrario solicita que se ingresen datos válidos.</p> <p>7. El sistema muestra un mensaje informado que se ingresó un nuevo estudiante</p>
--	---

Nuevo estudiante – fuente propia

**Tabla 13 CU Eliminar estudiante**

Aplicación	
Administrar estudiantes	
Descripción del caso de uso	
Nombre:	Eliminar estudiante
Actores	Administrador
Función:	Eliminar un estudiante del sistema
Descripción:	El administrador puede eliminar un estudiante de la base de datos del sistema
Precondición:	<p>Que el estudiante a eliminar, se encuentre en la base de datos de la aplicación.</p> <p>Saber o consultar el número de cedula del estudiante a eliminar</p> <p>El estudiante a eliminar no puede estar registrado en un grado</p>
Éxito	Se elimina el estudiante de la base de datos del sistema
Fracaso	No se elimina el estudiante la base de datos del sistema
Iniciador	El administrador da clic sobre la opción Estudiantes, Eliminar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Estudiante</li> <li>2. El sistema abre el formulario Administrar estudiantes</li> </ol>

	<ol style="list-style-type: none"> <li>3. El administrador da clic sobre la opción Eliminar</li> <li>4. El sistema abre un formulario para eliminar al estudiante deseado</li> <li>5. El administrador digita el número de cedula del estudiante a eliminar y da clic sobre la opción eliminar</li> <li>6. El sistema busca al estudiante mediante el número de cedula diligenciado si este se encuentra y no está registrado en un grado lo elimina</li> <li>7. El sistema muestra un mensaje informando que se eliminó el estudiante</li> </ol>
--	---

Eliminar estudiante – fuente propia

**Tabla 14 CU Actualizar estudiante**

Aplicación	
Administrar estudiantes	
Descripción del caso de uso	
Nombre:	Actualizar estudiante
Actores	Administrador
Función:	Actualiza los datos de un estudiante
Descripción:	El administrador podrá actualizar uno o varios datos de un estudiante registrado en la base de datos del sistema
Precondición:	<p>Que el estudiante se encuentre en la base de datos de la aplicación</p> <p>Buscar al estudiante al cual se le desean actualizar los datos</p>

Éxito	Se guarden los cambios realizados sobre el estudiante que se desea actualizar
Fracaso	No se guarden los cambios realizados sobre el estudiante que se desea actualizar
Iniciador	El administrador da clic sobre la opción Estudiantes, Actualizar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Estudiantes</li> <li>2. El sistema abre el formulario Administrar estudiantes</li> <li>3. El administrador da clic sobre la opción Actualizar</li> <li>4. El sistema habilita el formulario para actualizar los datos de un estudiante</li> <li>5. El administrador busca y selecciona el estudiante que desea actualizar</li> <li>6. El sistema habilita el formulario con los datos del estudiante a actualizar</li> <li>7. El administrador modifica uno o varios datos del estudiante que desea actualizar y da clic sobre la opción Guardar</li> <li>8. El sistema valida los nuevos datos ingresados (campos obligatorios y duplicidad en la identificación del estudiante). Si estos datos son válidos actualiza los datos del estudiante, de lo contrario solicita que se ingresen datos válidos.</li> </ol>

	9. El sistema muestra un mensaje informando que se actualizaron los datos del estudiante
--	--

**Tabla 15 CU Buscar estudiante**

Aplicación	
Administrar estudiantes	
Descripción del caso de uso	
Nombre:	Buscar estudiantes
Actores	Administrador
Función:	Buscar los estudiantes en la aplicación
Descripción:	Buscar los estudiantes que se encuentran registrados en la base de datos de la aplicación
Precondición:	El estudiante a buscar se encuentre registrado en la base de datos de la aplicación
Éxito	Que se encuentren los datos del estudiante buscado
Fracaso	Que no se encuentren los datos del estudiante buscado
Iniciador	Diligenciar los datos del estudiante a buscar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Estudiantes</li> <li>2. El sistema abre el formulario Administrar estudiantes</li> <li>3. El administrador da clic sobre la opción Buscar</li> <li>4. El sistema habilita el formulario para buscar un estudiante</li> <li>5. El administrador diligencia el número de cedula del estudiante a buscar y da clic sobre la opción buscar</li> </ol>

	6. El sistema muestra el estudiante que coincide con la búsqueda realizada
--	--

Buscar estudiante – fuente propia

**Tabla 16 CU Asignar grado**

Aplicación	
Administrar estudiantes	
Descripción del caso de uso	
Nombre:	Asignar grado
Actores	Administrador
Función:	Asignar grado al estudiante
Descripción:	Registrar al estudiante en determinado grado
Precondición:	Que el estudiante se encuentre registrado en la base de datos del sistema Que los grados, grupos se encuentren registradas en la base de datos del sistema
Éxito	Que se le asigne grado al estudiante seleccionado
Fracaso	Que no se le asigne grado al estudiante seleccionado
Iniciador	Seleccionar al estudiante
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador selecciona un estudiante</li> <li>2. El administrador selecciona un grado</li> <li>3. El sistema lista los grupos del grado seleccionado</li> <li>4. El administrador selecciona el grupo deseado y da clic sobre la opción Guardar</li> </ol>

	5. El sistema registra al estudiante en el grado, grupo seleccionado
--	--

Asignar grado – fuente propia

**Tabla 17 CU Nueva asignatura**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Nueva asignatura
Actores	Administrador
Función:	Registrar una asignatura en el sistema
Descripción:	El administrador podrá diligenciar los datos solicitados para ingresar una asignatura en la base de datos del sistema.
Precondición:	Haber iniciado sesión como administrador
Éxito	Se guarda la información de la asignatura en el sistema
Fracaso	No se guarda la información de la asignatura en el sistema
Iniciador	El administrador da clic sobre la opción Asignaturas
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Asignaturas</li> <li>2. El sistema abre el formulario (Administrar asignaturas)</li> <li>3. El administrador da clic sobre la opción Nuevo</li> <li>4. El sistema abre un nuevo formulario solicitando los datos de la asignatura</li> <li>5. El administrador ingresa los datos solicitados por el sistema y da clic sobre la opción Guardar</li> </ol>

	<p>6. El sistema valida los datos ingresados (campos obligatorios y duplicidad de código). Si los datos son válidos guarda la información de lo contrario solicita que se ingresen datos válidos.</p> <p>7. El sistema muestra un mensaje informado que se ingresó una nueva asignatura</p>
--	---

Nueva asignatura – fuente propia

**Tabla 18 CU Eliminar asignatura**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Eliminar asignatura
Actores	Administrador
Función:	Eliminar una asignatura del sistema
Descripción:	El administrador puede eliminar una asignatura de la base de datos del sistema
Precondición:	Que la asignatura a eliminar, se encuentre en la base de datos de la aplicación.  Saber o consultar el código de la asignatura a eliminar
Éxito	Se elimina la asignatura de la base de datos de la aplicación
Fracaso	No se elimina la asignatura de la base de datos de la aplicación
Iniciador	El administrador da clic sobre la opción Asignaturas, Eliminar



Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Asignaturas</li> <li>2. El sistema abre el formulario Administrar asignaturas</li> <li>3. El administrador da clic sobre la opción Eliminar</li> <li>4. El sistema abre un formulario para eliminar la asignatura deseada</li> <li>5. El administrador digita el código de la asignatura a eliminar y da clic sobre la opción eliminar</li> <li>6. El sistema busca la asignatura mediante el código diligenciado si este se encuentra la elimina, de lo contrario solicita que se digite un código valido</li> <li>7. El sistema muestra un mensaje informando que se eliminó la asignatura</li> </ol>
---------------------	--

Eliminar asignatura – fuente propia

**Tabla 19 CU Actualizar asignatura**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Actualizar asignatura
Actores	Administrador
Función:	Actualiza los datos de una asignatura
Descripción:	El administrador podrá actualizar los datos de una asignatura registrada en la base de datos del sistema

Precondición:	<p>Que la asignatura se encuentre en la base de datos de la aplicación</p> <p>Buscar la asignatura a la cual se le desean actualizar los datos</p>
Éxito	Se guarden los cambios realizados sobre la asignatura que se desea actualizar
Fracaso	No se guarden los cambios realizados sobre la asignatura que se desea actualizar
Iniciador	El administrador da clic sobre la opción Asignaturas, Actualizar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Asignaturas</li> <li>2. El sistema abre el formulario Administrar asignaturas</li> <li>3. El administrador da clic sobre la opción Actualizar</li> <li>4. El sistema habilita el formulario para actualizar los datos de una asignatura</li> <li>5. El administrador busca y selecciona la asignatura que desea actualizar</li> <li>6. El sistema habilita el formulario con los datos de la asignatura a actualizar</li> <li>7. El administrador modifica uno o varios datos de la asignatura que desea actualizar y da clic sobre la opción Guardar</li> <li>8. El sistema valida los nuevos datos ingresados (campos obligatorios y duplicidad en el código de la asignatura). Si</li> </ol>

	<p>estos datos son válidos actualiza los datos de la asignatura, de lo contrario solicita que se ingresen datos válidos.</p> <p>9. El sistema muestra un mensaje informando que se actualizaron los datos de la asignatura</p>
--	--

Actualizar asignatura – fuente propia

**Tabla 20 CU Buscar asignatura**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Buscar asignaturas
Actores	Administrador
Función:	Buscar las asignaturas en la aplicación
Descripción:	Buscar las asignaturas que se encuentran registradas en la base de datos de la aplicación
Precondición:	La asignatura a buscar se encuentre registrada en la base de datos de la aplicación
Éxito	Que se encuentren los datos de la asignatura buscada
Fracaso	Que no se encuentren los datos de la asignatura buscada
Iniciador	Diligenciar los datos de la asignatura a buscar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Asignaturas</li> <li>2. El sistema abre el formulario Administrar asignaturas</li> <li>3. El administrador da clic sobre la opción Buscar</li> </ol>

	<p>4. El sistema habilita el formulario para buscar una asignatura</p> <p>5. El administrador diligencia el código de la asignatura a buscar y da clic sobre la opción buscar</p> <p>6. El sistema muestra la asignatura que coincide con la búsqueda realizada</p>
--	---

Buscar asignatura – fuente propia

**Tabla 21 CU Nuevo grupo**

Aplicación	
Administrar grupo	
Descripción del caso de uso	
Nombre:	Nuevo grupo
Actores	Administrador
Función:	Registrar un nuevo grupo en el sistema
Descripción:	El administrador podrá ingresar un nuevo grupo en uno de los grados (1 - 11) existentes, diligenciando los datos solicitados por el sistema.
Precondición:	Haber iniciado sesión como administrador
Éxito	Se guarda la información del nuevo grupo en el sistema
Fracaso	No se guarda la información del grupo en el sistema
Iniciador	El administrador da clic sobre la opción Grados
Escenario principal	<p>1. El administrador da clic sobre la opción Grados</p> <p>2. El sistema abre el formulario (Administrar grados)</p>

	<p>3. El administrador diligencia los datos solicitados por el sistema para crear un nuevo grado y da clic sobre la opción Guardar</p> <p>4. El sistema valida los datos ingresados (campos obligatorios y duplicidad de código). Si los datos son válidos guarda la información de lo contrario solicita que se ingresen datos válidos.</p> <p>5. El sistema muestra un mensaje informado que se registró un nuevo grado</p>
--	---

Nuevo grupo – fuente propia

**Tabla 22 CU Eliminar grupo**

Aplicación	
Administrar grados	
Descripción del caso de uso	
Nombre:	Eliminar grupo
Actores	Administrador
Función:	Eliminar un grupo del sistema
Descripción:	El administrador podrá eliminar los registros de un grupo de la base de datos del sistema.
Precondición:	Haber iniciado sesión como administrador Que el grupo a eliminar se encuentre registrado
Éxito	Se elimina el grupo de la base de datos del sistema
Fracaso	No se elimina el grupo de la base datos del sistema

Iniciador	El administrador da clic sobre la opción Grados
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Grados</li> <li>2. El sistema abre el formulario (Administrar grados)</li> <li>3. El administrador selecciona el grado (1 - 11) que desea eliminar</li> <li>4. El sistema muestra los grados del grupo seleccionado</li> <li>5. El administrador selecciona el grupo que desea eliminar y da clic sobre la opción eliminar</li> <li>6. El sistema elimina el grupo seleccionado</li> <li>7. El sistema muestra un mensaje informando que se eliminó el grupo</li> </ol>

Eliminar grupo – fuente propia

**Tabla 23 CU Nuevo logro**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Nuevo logro
Actores	Administrador
Función:	Registrar un logro en el sistema
Descripción:	El administrador podrá diligenciar los datos solicitados para ingresar un logro en la base de datos del sistema.
Precondición:	Tener registrada la asignatura a la cual pertenece el logro
Éxito	Se guarda la información del logro en el sistema

Fracaso	No se guarda la información del logro en el sistema
Iniciador	El administrador da clic sobre la opción Logros
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador busca y selecciona la Asignatura y da clic sobre la opción Logros</li> <li>2. El sistema habilita el formulario para registrar un nuevo logro</li> <li>3. El administrador da clic sobre la opción Nuevo</li> <li>4. El sistema abre un nuevo formulario solicitando los datos del logro</li> <li>5. El administrador ingresa los datos solicitados por el sistema y da clic sobre la opción Guardar</li> <li>6. El sistema valida los datos ingresados (campos obligatorios y duplicidad de código). Si los datos son válidos guarda la información, de lo contrario solicita que se ingresen datos válidos.</li> <li>7. El sistema muestra un mensaje informado que se ingresó el nuevo logro</li> </ol>

Nuevo logro – fuente propia

**Tabla 24 CU Eliminar logro**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Eliminar logro

Actores	Administrador
Función:	Eliminar un logro del sistema
Descripción:	El administrador puede eliminar un logro de la base de datos del sistema
Precondición:	Que el logro a eliminar, se encuentre en la base de datos de la aplicación.  Saber o consultar el código del logro a eliminar
Éxito	Se elimina el logro de la base de datos de la aplicación
Fracaso	No se elimina el logro de la base de datos de la aplicación
Iniciador	El administrador da clic sobre la opción Logros
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador selecciona la asignatura del logro a eliminar y da clic sobre la opción logros</li> <li>2. El sistema abre el formulario Administrar logros</li> <li>3. El sistema lista los logros de la asignatura seleccionada</li> <li>4. El administrador selecciona el logro que desea eliminar y da clic sobre la opción eliminar</li> <li>5. El sistema eliminar el logro seleccionado</li> <li>6. El sistema muestra un mensaje informando que se eliminó el logro seleccionado</li> </ol>

Eliminar logro – fuente propia

**Tabla 25 CU Actualizar logro**

Aplicación
Administrar asignaturas



Descripción del caso de uso	
Nombre:	Actualizar logros
Actores	Administrador
Función:	Actualiza los datos de un logro
Descripción:	El administrador podrá actualizar los datos de un logro registrado en la base de datos del sistema
Precondición:	Que el logro se encuentre en la base de datos de la aplicación Buscar la asignatura a la cual pertenece el logro que se desea actualizar
Éxito	Se guarden los cambios realizados sobre el logro que se desea actualizar
Fracaso	No se guarden los cambios realizados sobre el logro que se desea actualizar
Iniciador	El administrador selecciona la asignatura y da clic sobre la opción Logros
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador selecciona la asignatura del logro a actualizar y da clic sobre la opción logros</li> <li>2. El sistema abre el formulario Administrar logros</li> <li>3. El administrador da clic sobre la opción Actualizar</li> <li>4. El sistema habilita el formulario para actualizar los logros de la asignatura seleccionada</li> <li>5. El sistema lista los logros de la asignatura seleccionada</li> </ol>

	<p>6. El administrador busca y selecciona el logro que desea actualizar</p> <p>7. El sistema habilita el formulario con los datos del logro a actualizar</p> <p>8. El administrador modifica los datos del logro que desea actualizar y da clic sobre la opción Guardar</p> <p>9. El sistema valida los nuevos datos ingresados (campos obligatorios y duplicidad en el código del logro). Si estos datos son válidos actualiza los datos del logro, de lo contrario solicita que se ingresen datos válidos.</p> <p>10. El sistema muestra un mensaje informando que se actualizaron los datos del logro</p>
--	--

Actualizar logro – fuente propia

**Tabla 26 CU Buscar logro**

Aplicación	
Administrar asignaturas	
Descripción del caso de uso	
Nombre:	Buscar logros
Actores	Administrador
Función:	Buscar los logros en la aplicación
Descripción:	Buscar los logros que se encuentran registradas en la base de datos de la aplicación

Precondición:	Los logros a buscar se encuentren registrados en la base de datos de la aplicación
Éxito	Que se encuentren los datos del logro consultado
Fracaso	Que no se encuentren los datos del logro consultado
Iniciador	Diligenciar los datos del logro a buscar
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre la opción Logros</li> <li>2. El sistema abre el formulario Administrar logros</li> <li>3. El administrador da clic sobre la opción Buscar</li> <li>4. El sistema habilita el formulario para buscar un logro</li> <li>5. El administrador diligencia el código del logro a consultar y da clic sobre la opción buscar</li> <li>6. El sistema muestra los logros que coincide con la búsqueda realizada</li> </ol>

Buscar logro – fuente propia

**Tabla 27 CU Registrar notas**

Aplicación	
Registrar notas	
Descripción del caso de uso	
Nombre:	Registrar notas
Actores	Docente
Función:	Registrar nota a los estudiantes

Descripción:	El docente podrá registrarle notas a los estudiantes en los distintos grados, grupos y respectivas asignaturas en las cuales dicta clases
Precondición:	El docente haya iniciado sesión El docente tenga registrado carga docente (Grados, grupos, asignaturas) Haya estudiantes registrados en los grupos donde el docente dicta clases
Éxito	Se registren notas a los estudiantes
Fracaso	No se registren notas a los estudiantes
Iniciador	Iniciar sesión como docente
Escenario principal	<ol style="list-style-type: none"> <li>1. Después de haber iniciado sesión como docente</li> <li>2. El sistema solicita en el formulario Grado y Grupo en los cuales dicta clases el docente registrado</li> <li>3. El docente selecciona el Grado en el cual va a calificar las notas</li> <li>4. El sistema lista los grupos en los cuales dicta clases el docente registrado</li> <li>5. El docente selecciona el grupo en el cual va a registrar las notas</li> <li>6. El sistema lista los estudiantes que se encuentran registrados en el Grado, grupo seleccionados</li> </ol>

	<p>7. El sistema solicita las asignaturas que dicta el docente registrado en el grado, grupo seleccionado</p> <p>8. El docente selecciona la asignatura en la cual va a registrar las notas</p> <p>9. El sistema lista los periodos</p> <p>10. El docente selecciona el periodo en el cual va a registrar las notas</p> <p>11. El docente selecciona el estudiante al cual le va a registrar las notas</p> <p>12. El sistema habilita el formulario para registrar la nota al estudiante seleccionado</p> <p>13. El docente registra la nota y da clic sobre la opción Guardar</p> <p>14. El sistema guarda la nota del estudiante seleccionado</p> <p>15. El sistema muestra un mensaje informando que la nota se registró exitosamente.</p>
--	---

Registrar notas – fuente propia

**Tabla 28 CU Ver notas**

Aplicación	
Ver notas	
Descripción del caso de uso	
Nombre:	Ver notas
Actores	Estudiante

Función:	Ver las notas registradas
Descripción:	Ver las notas que tienen registrados los estudiantes en las diferentes asignaturas en cada uno de los periodos
Precondición:	El estudiante haya iniciado sesión
Éxito	Se muestran las notas del estudiante registrado
Fracaso	No se muestran las notas del estudiante registrado
Iniciador	Iniciar sesión como un estudiante
Escenario principal	<ol style="list-style-type: none"> <li>1. Después de haber iniciado sesión como estudiante</li> <li>2. El sistema solicita el periodo del cual se desean revisar las notas</li> <li>3. El estudiante selecciona un periodo</li> <li>4. El sistema lista las asignaturas que ve el estudiante y muestra las notas en las distintas asignaturas que tiene el estudiante en el periodo seleccionado.</li> </ol>

Ver notas – fuente propia

**Tabla 29 CU Generar reportes**

Aplicación	
Generar reportes	
Descripción del caso de uso	
Nombre:	Generar reportes
Actores	Administrador
Función:	Generar distintos reportes

Descripción:	El administrador podrá generar los distintos tipos de reportes con los que cuenta la aplicación
Precondición:	Haber iniciado sesión como administrador
Éxito	Se muestran los reportes de acuerdo a los criterios seleccionados
Fracaso	No se muestran los reportes deseados
Iniciador	Iniciar sesión como administrador y dar clic en la opción reportes
Escenario principal	<ol style="list-style-type: none"> <li>1. El administrador da clic sobre opción reportes</li> <li>2. El sistema habilita el formulario para crear el reporte deseado de acuerdo al criterio seleccionado</li> <li>3. El administrador selecciona el criterio del reporte</li> <li>4. El sistema genera el reporte de acuerdo al criterio seleccionado</li> </ol>

Generar reportes – fuente propia

### 7.1.2.3 Campos base de datos

*Tabla 30 - Tabla Docente*

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_docente	doc_identificacion		identificacion	Int	Almacena el número de identificación del docente debe ser único
			nombre	Nvarchar(50)	Almacena el nombre del docente
			apellido1	Nvarchar(50)	Almacena el primer apellido del docente
			apellido2	Nvarchar(50)	Almacena el segundo apellido del docente
			genero	Nchar(10)	Almacena el género del docente (Hombre, Mujer)



			pass	Nvarchar(20)	Almacena la contraseña del docente
			tipo	Int	Almacena el tipo de usuario del docente (Administrador, Docente)
			correo	Nvarchar(50)	Campo destinado para el correo del docente

Fuente propia

**Tabla 31 - Tabla Logros**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_logros	log_codigo		codigo	Int	Almacena el código del logro debe ser único
			descripcion	Nvarchar(MAX)	Almacena la descripción del logro

			tipo	Nchar(10)	Almacena el tipo de logro (Fortaleza, Dificultad)
		*	asig_codigo	Nchar(10)	Especifica la asignatura a la cual pertenece el logro

Fuente propia

**Tabla 32 - Tabla Docente Grupo**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_doc_grup	doc_grup_codigo		codigo	Int	Almacena la llave primaria del docente que dicta una asignatura en determinado grado
		*	doc_identificacion	Int	Almacena el número de identificación del docente que dicta la

					asignatura en determinado grado
		*	asig_codigo	Nchar(10)	Almacena el código de la asignatura que dicta el docente en determinado grado
		*	gru_codigo	Nchar(10)	Almacena el código del grado en el cual el docente dicta determinada asignatura

Fuente propia

**Tabla 33 - Tabla Asignatura**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_asignatura	asig_codigo		codigo	Nchar(10)	Almacena el código de la asignatura

			nombre	Nchar(20)	Almacena el nombre de la asignatura
--	--	--	--------	-----------	-------------------------------------

Fuente propia

**Tabla 34 - Tabla Grupo**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_grupo	gru_codigo		codigo	Nchar(10)	Almacena el código del grupo
		*	grad_codigo	Int	Almacena el grado del grupo (1 - 11)
			grupo	Nchar(10)	Almacena el grupo (a, b, c...)
			descripcion	Nvarchar(50)	Almacena una pequeña descripción del grado

Fuente propia

**Tabla 35 - Tabla Grado**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_grado	grad_codigo		código	Int	Almacena el código del grado (1 - 11)
			descripcion	Nvarchar(50)	Almacena el nombre del grado en letras

Fuente propia

**Tabla 36 - Tabla Notas**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_notas	not_codigo		código	Int	Almacena la llave primaria de la nota
			nota	Decimal (3,2)	Almacena la nota del estudiante

		*	asig_codigo	Nchar(10)	Almacena el código de la asignatura a la cual pertenece la nota
		*	est_identificación	Int	Almacena el número de identificación del estudiante
		*	per_codigo	Int	Almacena el código del periodo que se está calificando
			descripción	Nvarchar(MAX)	Almacena una pequeña descripción de la nota que se está calificando

Fuente propia

**Tabla 37 - Tabla Estudiante Grupo**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
-------	-------------	-------------	--------	------	-------------

tbl_est_grup	est_grup_codigo		codigo	Int	Almacena el código (llave primaria) del grupo
		*	est_identificacion	Int	Almacena el código del estudiante que pertenece a ese grupo
		*	gru_codigo	Nchar(10)	Almacena el código del grupo al que pertenece el estudiante
		*	grad_codigo	Int	Almacena el código del grado al que pertenece el estudiante

Fuente propia

**Tabla 38 - Tabla Estudiante**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
-------	-------------	-------------	--------	------	-------------

tbl_estudiante	est_identificacion		identificacion	Int	Almacena el número de identificación del estudiante
			nombre	Nvarchar(50)	Almacena el nombre del estudiante
			apellido1	Nvarchar(50)	Almacena el primer apellido del estudiante
			apellido 2	Nvarchar(50)	Almacena el segundo apellido del estudiante
			genero	Nchar(10)	Almacena el género del estudiante (Hombre Mujer)
			pass	Nvarchar(20)	Almacena la contraseña del estudiante



			correo	Nvarchar(50)	Almacena el correo electrónico del estudiante
			Nombre_acudiente	Nvarchar(50)	Almacena el nombre del acudiente
			Apellido_acudiente	Nvarchar(50)	Almacena los apellidos del acudiente
			Id_acudiente	Int	Almacena el numero de cedula del acudiente
			Tel_acudiente	int	Alacena el número telefónico del acudiente

Fuente propia

**Tabla 39 - Tabla Periodo**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_periodo	per_codigo		codigo	Int	Almacena el código del periodo (1 - 4)

			descripcion	Nvarchar(50)	Almacena una pequeña descripción del periodo
--	--	--	-------------	--------------	--

Fuente propia

**Tabla 40 Tabla Tipo usuario**

TABLA	PRIMARY KEY	FOREIGN KEY	CAMPOS	TIPO	DESCRIPCIÓN
tbl_tipo_usuario	usu_codigo		Usu_codigo	Int	Almacena el código del tipo de usuario, que puede ser (1 o 2)
			Usu_tipo	Nchar(20)	Almacena una pequeña descripción del tipo de usuario que puede ser 1 Administrador, 2 docente

Fuente propia

## 7.1.3 FASE DE CONSTRUCCIÓN

### 7.1.3.1 Diagrama entidad relación

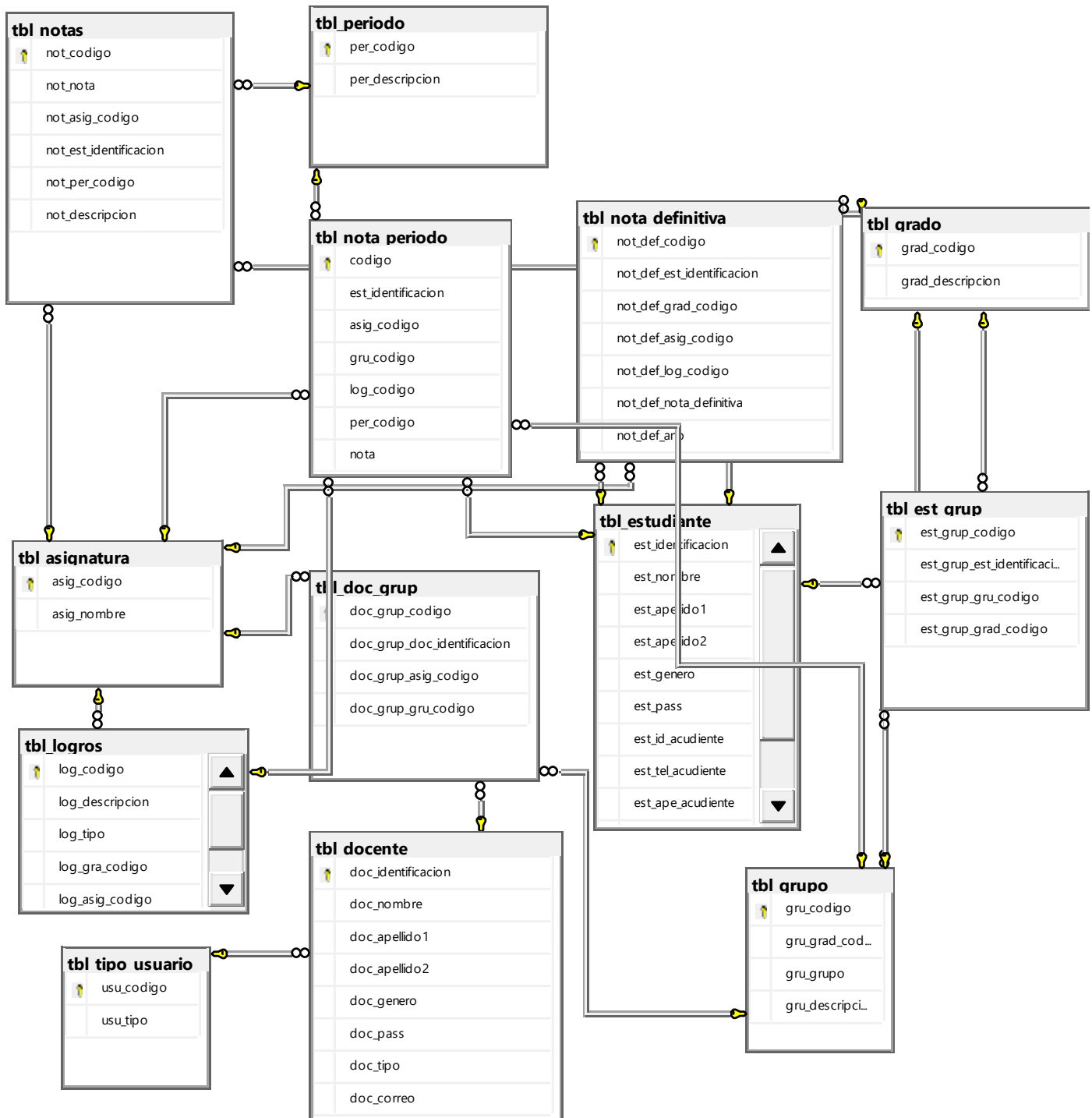


Ilustración 20 Diagrama Entidad Relación—fuente propia

### 7.1.3.2 Programación

Durante esta fase se seleccionaron e implementaron los casos de uso considerados principales para el funcionamiento de la aplicación, posteriormente en siguientes iteraciones del desarrollo de la aplicación, se fueron realizando los demás Casos de uso propuestos obteniendo como resultado una versión estándar y funcional de la aplicación.

### 7.1.3.3 Interfaces de usuario

Con el objetivo de que la aplicación cumpla con los requerimientos del usuario se procede a implementar una interfaz que cuente con un gran índice de usabilidad, facilitando en gran medida la interacción del usuario final con la aplicación, así como la utilización de colores suaves y agradables a la vista los cuales la hacen un poco más llamativa.

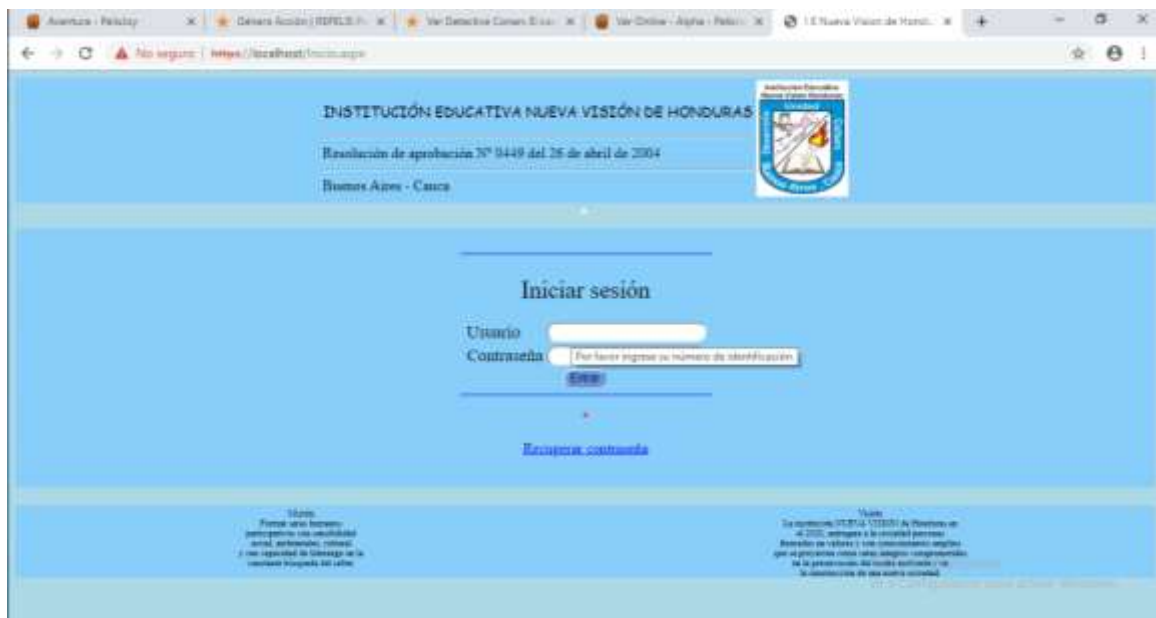


Ilustración 21 IU Iniciar sesión– fuente propia



Ilustración 22 IU Insertar grados–fuente propia



Ilustración 23 IU Insertar notas–fuente propia



Ilustración 24 IU Insertar nota individual– fuente propia



Ilustración 25 IU Notas estudiante– fuente propia

### 7.1.3.4 Pruebas

Durante el desarrollo de la aplicación se han ido realizando pruebas con el objetivo de medir la funcionalidad y evitar posibles fallos críticos de la aplicación, además de contar con una interfaz agradable y fácil de utilizar para el usuario final.

Después de haber finalizado el desarrollo de todos los casos de uso se prosiguió a realizar pruebas que garanticen el buen funcionamiento de la aplicación.

Durante el desarrollo de estas pruebas se encontraron errores de validación como el ingreso de registros duplicados, es de vital importancia mencionar que en su mayoría se encontraron errores de ingreso de datos en los cuales la aplicación permitía el ingreso de caracteres en campos donde van exclusivamente valores numéricos, obteniendo como requerimiento la corrección de estos fallos.

## **7.1.4 FASE DE TRANSICIÓN**

### **7.1.4.1 Entrega del producto**

Con el propósito de realizar la entrega del producto final (Aplicación para el registro, control y consulta de notas en la institución educativa nueva visión de honduras) se realiza la instalación de la aplicación con el montaje de la respectiva base de datos, garantizando y probando al usuario final el óptimo funcionamiento de ésta en la red de computadores de la institución educativa.

### **7.1.4.2 Capacitación al usuario final**

Para realizar la capacitación de la aplicación al usuario final se toma una jornada de trabajo demostrando y explicando las funcionalidades de la aplicación en óptimo funcionamiento mediante el uso del manual de usuario.

### **7.1.4.3 Manual de usuario**

Se entrega un manual de usuario de la aplicación en el cual se muestran las funcionalidades de la aplicación, logrando de esta manera una fácil utilización de la aplicación y soporte en caso de requerirlo.

## **8 CAPÍTULO V - RESULTADOS**

### **8.1 RESULTADOS**

Como principal resultado de este trabajo se obtuvo una aplicación web para el control, registro y consulta de notas para institución educativa nueva visión de honduras, con su respectiva base de datos.

La implementación del aplicativo web para el control, registro y consulta de notas, dio a la institución educativa nueva visión de honduras una manera diferente de almacenar la información, garantizando la portabilidad y seguridad de la información, mediante la utilización de una base de datos. Además de que los directivos de esta institución educativa pueden ver el rendimiento de los estudiantes de una forma más fácil y ágil, permitiendo tomar acciones rápidas en caso de que los alumnos tengan un bajo rendimiento académico.



## **9 CAPÍTULO VI - CONCLUSIONES**

### **9.1 CONCLUSIONES**

Se logró brindar una solución tecnológica a la problemática que tenía la institución educativa nueva visión de honduras en el proceso de registro, control y consulta de notas. Sistematizando estos datos, para así conservar registros históricos con el transcurso del tiempo.

Mediante la base de datos de la aplicación se logra que la información de las notas de los estudiantes se encuentre almacenada de una forma segura y confiable

Mediante el desarrollo de este proyecto se logró un mejor tratamiento de los datos, así como la portabilidad de la información

Las pruebas realizadas a la aplicación garantizaron el óptimo funcionamiento de cada una de sus partes, logrando obtener el producto esperado

### **9.2 RECOMENDACIONES**

Se recomienda realizarle mejoras en los servicios que software de registro, control y consulta de notas presta, incorporando nuevas funciones que sean útiles a la comunidad educativa.

Se sugieren realizarle mantenimiento periódico a la aplicación para garantizar su óptimo funcionamiento, y no almacenar archivos innecesarios.

## 10 BIBLIOGRAFÍA

ARIAS, Ángel. Aprende a programar ASP .NET y C# 2ª ed.

ARIAS CHAVES, Michael. La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software

ÁREA MOREIRA, Manuel. La integración escolar de las nuevas tecnologías. Entre el deseo y la realidad. Director del Laboratorio de Educación y Nuevas Tecnologías EDULLAB. Universidad de La Laguna.

BAYONA ZUBIETA, Jenny Carolina, PINEDA SAMACÁ, Olga Lucía y PARDO MAHECHA, Oscar Daniel. El papel de la Ingeniería de Software en el desarrollo de aplicaciones.

BERZAL GALIANO, Fernando y CORTIJO BON, Francisco. Curso de C#. La plataforma .Net. Disponible en: <http://elvex.ugr.es/decsai/csharp/dotnet/index.xml>

BERZAL, Fernando, CORTIJO, Francisco José y CUBERO, Juan Carlos. Desarrollo Profesional de Aplicaciones Web con ASP .NET

BOOCH, Graby; RUMBAUGH, Jim, JACOBSON, Ivar. El lenguaje unificado de modelado.

CAPÍTULO 2: Ingeniería de Software, Análisis y Diseño. Disponible en: [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/fuentes\\_k\\_jf/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf)

CEBALLOS VILLACH, Jordi. Introducción a .NET

CEBALLOS, Francisco Javier. Microsoft Visual Basic .NET Lenguaje y aplicaciones. 3ª ed.

CIUDAD EDUCATIVA. Portafolio 2018. Disponible en: <https://www.ciudadeducativa.com/ciudadeducativa-colegio.pdf>

COLOMBIA. Ministerio de Educación Nacional. Decreto 1860 de 1994 (agosto 3). “Por el cual se reglamenta parcialmente la Ley 115 de 1994, en los aspectos pedagógicos y organizativos generales”. Diario oficial No 41.473, del 5 de agosto de 1994.

COLWEB. Sistema de Gestión Académica. Disponible en:  
<http://www.colweb.com.co/nosotros.html>

Disponible en: [ftp://ciclope.unicauca.edu.co/ingenieria\\_web/web-book-a4-ASPNET.pdf](ftp://ciclope.unicauca.edu.co/ingenieria_web/web-book-a4-ASPNET.pdf)

DOMÍNGUEZ ALFONSO, Rocío. Nuevas Tecnologías y Educación en el siglo XXI. Universidad de Granada.

DRAKE, J. M. Programación orientada a objetos: Lenguajes, Metodologías y Herramientas Master de Computación.

ELSTEIN, Silvia. Nuevas tecnologías y educación. Departamento de Lenguas, Facultad de Ciencias Humanas, Universidad Nacional de Río Cuarto.

FERGARCIA. Entorno de desarrollo integrado IDE. Programación orientada a objetos. 2013. Disponible en: <https://fergarcia.wordpress.com/2013/01/25/entorno-de-desarrollo-integrado-ide/>

FERNÁNDEZ PRIETO, Manuel Santiago. Las nuevas tecnologías en la educación. Disponible en: <http://www.uhu.es/cine.educacion/didactica/0071tecnologiaaulas.htm>

GARRIDO PINO, Miguel Ángel. Evaluación comparativa de aplicaciones Web entre JJ2EE y Microsoft.Net. (Tesis). Universidad Católica de Temuco. Facultad de Ingeniería. Temuco. 2006. Disponible en: <http://www.issi.uned.es/pea/programacion-c/downloads/tesis.pdf>

GIL. Gustavo Daniel. Herramienta para implementar LEL y escenarios (TILS)

GUALIX, Javier. Así es Visual Studio 2017 el IDE perfecto para crear APPS, Windows, IOS Android y la nube. Visual Studio. 2017. Disponible en: <https://www.microsoftinsider.es/121003/asi-visual-studio-2017-ide-perfecto-crear-apps-windows-ios-android-la-nube/>

JORDISAN.NET. ¿Qué es un framework? Disponible en: <https://jordisan.net/blog/2006/que-es-un-framework/>

KAWULICH, Barbara B. La observación participante como método de recolección de datos.

LETELIER, P. Proceso de desarrollo de software.

LINARES I PELLICER, Jordi. Programación amb C# .NET Disponible en:  
<http://users.dsic.upv.es/~jlinares/csharp/Tema%201.pdf>

MAGAZINE. Metodología RUP y Ciclo de Vida. Disponible en:  
<http://rupmetodologia.blogspot.com/>

MARTÍNEZ, Alejandro y MARTÍNEZ, Raúl. Guía a Rational Unified Process.

MARTIN ESCOFET, Carmen. El lenguaje SQL.

MDN web docs. HTML. 2017. Disponible en: <https://developer.mozilla.org/es/docs/Web/HTML>

MERELO GUERVOS, Juan Julián. Introducción a los servicios Web y Microsoft.Net. 2002.  
Disponible en: <http://geneura.ugr.es/~jmerelo/ws/>

MICROSOFT. Introducción a .NET Framework. Disponible en: <https://docs.microsoft.com/es-es/dotnet/framework/get-started/>

NOVEDADES DE Visual Studio 2017. 2018. Disponible en: <https://docs.microsoft.com/es-es/visualstudio/ide/whats-new-in-visual-studio?view=vs-2017>

OLGUIN ESPINOZA, José Martín. Análisis orientado a objetos. 2004. Disponible en:  
<http://yaqui.mx1.uabc.mx/~molguin/as/IngSoft%201-4.pdf>

OK HOSTING. Herramientas de desarrollo de software. Disponible en:  
<https://okhosting.com/blog/herramientas-de-desarrollo-de-software/>

PAYNE, Chris. Aprendiendo .NET en 21 Lecciones avanzadas

PÉREZ A., Oiver Andrés. Cuarto enfoques metodológicos para el desarrollo de software RUP – MSF – XP – SCRUM.

PRESSMAN S. Roger. Ingeniería de software: un enfoque practico. Unidad 1. Disponible en:  
<http://www.itlalaguna.edu.mx/academico/carreras/sistemas/ingsoftware1/Unidad1.pdf>

SOMMERVILLE, Ian. Ingeniería de software

SCHOOL PACK. Sistema de información integral para instituciones educativas. Cundinamarca, Colombia. 2000. Disponible en:

[http://www.upwaresoft.com/flipbookSchool/pages/school\\_pack.pdf](http://www.upwaresoft.com/flipbookSchool/pages/school_pack.pdf)

VARGAS DEL VALLE, Ricardo y MALTÉS GRANADOS, Juan. Programación en capas. Universidad de Costa Rica. Disponible en: <http://www.di-mare.com/adolfo/cursos/2007-2/pp-3capas.pdf>