

ANÁLISIS Y DISEÑO DEL SOFTWARE PARA EL CONTROL DE ASISTENCIA  
DE ESTUDIANTES.

“STEEL”

JUAN CARLOS ALONSO BERNATE

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS

FACULTAD DE INGENNIERIA

TECNOLOGÍA EN INFORMÁTICA

SOACHA

2012

ANÁLISIS Y DISEÑO DE SOFTWARE PARA EL CONTROL DE ASISTENCIA DE  
ESTUDIANTES,

“ STEEL ”

JUAN CARLOS ALONSO BERNATE

Trabajo de grado presentado como requisito para optar al título de  
Tecnólogo en Informática

Docente, LEONARDO ENRIQUE PINZÓN CABALLERO  
Licenciado

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS

FACULTAD DE INGENNIERIA

TECNOLOGÍA EN INFORMÁTICA

SOACHA

2012

Nota de aceptación:

---

---

---

---

---

---

---

---

---

---

---

Firma del presidente del jurado

---

Firma del jurado

---

Firma del jurado

Soacha 21 de Junio del 2012

## **DEDICATORIA**

A mi familia quien me apoyo en cada momento, a mi esposa que es la fuerza que me ayuda a seguir a adelante, a mi hijo que es mi vida entera, y a Dios por darme la sabiduría, entendimiento y ciencia para realizar este gran logro.

## **AGRADECIMIENTOS**

Agradezco a Dios, mis padres, hermanos, mi esposa y mi hijo que me acompañaron en cada una de las etapas de este trabajo y me dieron el valor para seguir adelante y no desfallecer.

De la misma manera al docente LEONARDO ENRIQUE PINZÓN CABALLERO por su paciencia, confianza en mí y la dirección de éste trabajo, al docente PABLO FONSECA por sus consejos, el apoyo y el ánimo que me brindo en este proceso, ala universidad y demás docentes que estuvieron prestos a brindar la mejor colaboración para conseguir este gran logro.

Gracias también a mis compañeros que me apoyaron y me permitieron compartir en sus vidas durante estos semestres.

## TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	18
1 PROBLEMA DE INVESTIGACIÓN.	19
1.1 TITULO DEL PROYECTO.	19
1.2 ORIGEN DEL PROYECTO	19
1.3 JUSTIFICACIÓN.	19
1.4 PLANTEAMIENTO DEL PROBLEMA.	20
1.4.1 Descripción del problema.	20
1.4.2 Formulación del problema.	20
1.5 OBJETIVOS.	22
1.5.1 Objetivo general.	22
1.5.2 Objetivo específico.	22
2. MARCO REFERENCIAL.	23
2.1 MARCO TEÓRICO.	23
2.2 Metodología aplicada Métrica V3.	23
2.3 MARCO TÉCNICO.	28
2.3.1 Normalización de base de datos.	28
2.3.2 PHP 5.	29
2.3.3 MYSQL.	31
2.3.4 HTML.	31
2.3.5 XAMPP.	32
2.4 ANTECEDENTES.	32
2.4.1 EasyClocking.	33
2.4.2 FS Acceso.	33
2.4.3 SquareNet.	33
2.4.4 Biopoint.	34

3. CONSIDERACIÓN DE DISEÑO.	35
3.1 REQUERIMIENTOS TÉCNICOS.	35
3.2 FACTIBILIDAD OPERATIVA.	35
3.3 FACTIBILIDAD ECONÓMICA.	36
3.4 ALCANCES DEL SISTEMA.	37
4. DISEÑO DEL SISTEMA.	39
4.1 DESCRIPCIÓN DEL SISTEMA.	39
4.2 ARQUITECTURA DEL SISTEMA.	39
4.3 MODELO DE CASOS DE USO.	40
4.3.1 Descripción general de los Actores.	42
4.3.2 Diagramas del modelo de Casos de Uso.	42
4.3.3 Modelos Objetos.	55
4.3.4 Modelos Dinámicos.	61
5. CONCLUSIONES.	69
6. RECOMENDACIONES Y PROPUESTAS DE MEJORAS FUTURAS	70
BIBLIOGRAFÍA.	71

## LISTA DE TABLAS

	Pág.
<b>TABLA 1.</b> Personal del proyecto.	36
<b>TABLA 2.</b> Costos del tiempo del personal del proyecto.	36
<b>TABLA 3.</b> Costo de materiales del proyecto.	36
<b>TABLA 4.</b> Costos varios del proyecto.	37
<b>TABLA 5.</b> Arquitectura física del sistema.	39
<b>TABLA 6.</b> Actores y casos de uso.	40
<b>TABLA 7.</b> Casos de uso para el registro del usuario.	43
<b>TABLA 8.</b> Casos de uso ingreso de usuario.	44
<b>TABLA 9.</b> Casos de uso registro de pagos.	45
<b>TABLA 10.</b> Casos de uso asistencia a clase.	46
<b>TABLA 11.</b> Casos de uso consulta de datos alumno.	47
<b>TABLA 12.</b> Casos de uso consulta de datos por alumno.	48
<b>TABLA 13.</b> Casos de uso consulta de asistencia (administrador).	49
<b>TABLA 14.</b> Casos de uso consulta de pago (administrador).	50
<b>TABLA 15.</b> Casos de uso consulta de grado del alumno (administrador).	51
<b>TABLA 16.</b> Casos de uso modificación de datos (administrador).	52
<b>TABLA 17.</b> Casos de uso eliminar datos (administrador).	52
<b>TABLA 18.</b> Casos de uso.	52
<b>TABLA 19.</b> Atributos de la tabla alumnos.	55
<b>TABLA 20.</b> Atributos de la tabla pagos.	57
<b>TABLA 21.</b> Atributos de la tabla asistencia.	57
<b>TABLA 22.</b> Atributos de la tabla usuarios.	58

**TABLA 23.**Atributos de la tabla graduación.

59

**TABLA 24.**Atributos de la tabla sedes.

## LISTA DE FIGURAS

	Pág.
<b>FIGURA 1.</b> Diagrama casos de uso, escenario general	41
<b>FIGURA 2.</b> Caso de uso, registro de usuario.	42
<b>FIGURA 3.</b> Caso de uso, ingreso de usuario.	43
<b>FIGURA 4.</b> Caso de uso, registro de pagos.	44
<b>FIGURA 5</b> Caso de uso, registro de asistencia a clase.	45
<b>FIGURA 6.</b> Caso de uso, consulta de datos de alumnos.	46
<b>FIGURA 7.</b> Caso de uso, consulta de datos por alumno.	47
<b>FIGURA 8.</b> Caso de uso, consulta de asistencia a clase.	48
<b>FIGURA 9.</b> Caso de uso, consulta de pagos.	49
<b>FIGURA 10.</b> Caso de uso, consulta de grado de alumnos.	50
<b>FIGURA 11.</b> Caso de uso, modificar datos.	51
<b>FIGURA 12.</b> Caso de uso, eliminar datos.	52
<b>FIGURA 13.</b> Caso de uso, generar reportes estadísticos.	53
<b>FIGURA 14.</b> Diagrama de clases, escenario general	54
<b>FIGURA 15.</b> Diagrama de clases, alumno.	55
<b>FIGURA 16.</b> Diagrama de clases, pago.	56
<b>FIGURA 17.</b> Diagrama de clases, asistencia.	57
<b>FIGURA 18.</b> Diagrama de clases, usuario.	58
<b>FIGURA 19.</b> Diagrama de clases, graduación.	59
<b>FIGURA 20.</b> Diagrama de clases, sedes.	59
<b>FIGURA 21.</b> Diagrama de secuencia, registro alumno.	61
<b>FIGURA 22.</b> Diagrama de secuencia, asistencia a clase.	61

<b>FIGURA 23.</b> Diagrama de secuencia, pagos de clase.	62
<b>FIGURA 24.</b> Diagrama de secuencia, actualizar registró alumno.	62
<b>FIGURA 25.</b> Diagrama de secuencia, consultar asistencia a clase.	63
<b>FIGURA 26.</b> Diagrama de secuencia, consultar .grado de alumno	64
<b>FIGURA 27.</b> Diagrama de secuencia, consultar registro de alumno.	64
<b>FIGURA 28.</b> Diagrama de secuencia, consultar pago de clases.	65
<b>FIGURA 29.</b> Diagrama de secuencia, eliminar registro de alumno.	66
<b>FIGURA 30.</b> Diagrama de secuencia, modificar registro de alumno.	66
<b>FIGURA 31.</b> Diagrama de secuencia, modificar pago de clase.	67

## GLOSARIO

**ACADEMIA:** Lugar adecuado para entrenamiento físico y mental.

**APLICACIONES WEB:** Herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

**ASP:** Framework para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y web XML. Apareció en enero de 2002 con la versión 1.0 del .NET Framework, y es la tecnología sucesora de la tecnología Active Server Pages (ASP).

**BASE DE DATOS:** Una base de datos o banco de datos (en ocasiones abreviada con la sigla *BD* o con la abreviatura *b. d.*) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.

**BASE DE DATOS RELACIONALES:** Base de datos que cumple con el modelo relacional, el cual es el modelo más utilizado en la actualidad para implementar bases de datos ya planificadas. Permiten establecer interconexiones (relaciones) entre los datos (que están guardados en tablas), y a través de dichas conexiones relacionar los datos de ambas tablas, de ahí proviene su nombre: "Modelo Relacional".

**BYTE:** Secuencia de bits contiguos, cuyo tamaño depende del código de información o código de caracteres en que sea definido.

**BATIZADO TROCA DE CORDAS:** Evento de la capoeira que se organiza cada año para cambiar de cinturón ascendiendo a un nivel superior según el tiempo del alumno y su experiencia.

**CACHE:** La caché de CPU, es una caché usada por la unidad central de procesamiento de una computadora para reducir el tiempo de acceso a la memoria. La caché es una memoria más diminuta y rápida, la cual almacena copias de datos ubicados en la memoria principal que se utilizan con más frecuencia.

**CAPOEIRA:** Arte marcial afro brasilero oculto en la danza.

**CLASE:** En la programación orientada a objetos, una clase es una construcción que se utiliza como un modelo (o plantilla) para crear objetos de ese tipo. El modelo describe el estado y el comportamiento que todos los objetos de la clase comparten.

**CORDA:** Cuerda que clasifica a los alumnos de capoeira según el nivel de formación.

**HOST:** Computadoras conectadas a una red, que proveen y utilizan servicios o de ella. Los usuarios deben utilizar *anfitriones* para tener acceso a la red.

**INTERFAZ DE USUARIO:** La interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.

**LICENCIA GNU:** Licencia creada por la Free Software Fundación en 1989 (la primera versión, escrita por Richard Stallman), y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

**MESUALIDAD:** pago realizado mensualmente para poder acceder a las clases de capoeira.

**NAMED PIPES:** Tubería nombrada (*named pipe* en inglés), también llamada FIFO por su comportamiento, es una extensión del concepto tradicional de tuberías utilizado en los Sistemas operativos POSIX, y es uno de los métodos de Comunicación entre procesos (IPC). Este concepto también se encuentra en Windows, si bien implementado con otra semántica.

**OFUSCACIÓN:** Se refiere al acto deliberado de realizar un cambio no destructivo, ya sea en el código fuente de un programa informático o código máquina cuando el programa está en forma compilada o binaria, con el fin de que no sea fácil de entender o leer.

**POO:** La programación orientada a objetos o POO (OOP según sus siglas en inglés) es un paradigma de programación que usa objetos y sus interacciones, para diseñar aplicaciones y programas informáticos. Está basado en varias técnicas, incluyendo herencia, abstracción, polimorfismo y encapsulamiento.

**REDUNDANCIA:** En teoría de la información, la redundancia es una propiedad de los mensajes, consistente en tener partes predictibles a partir del resto del mensaje y que por tanto en sí mismo no aportan nueva información o "repiten" parte de la información.

**SERVIDOR:** Computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.

**SCRIPTS (LENGUAJE INTERPRETADO):** Lenguaje de programación que está diseñado para ser ejecutado por medio de un intérprete, en contraste con los lenguajes compilados.

**SISTEMA DE INFORMACIÓN:** Un sistema de información (SI) es un conjunto de elementos orientados al tratamiento y administración de datos e información,

organizados y listos para su uso posterior, generados para cubrir una necesidad u objetivo.

**SISTEMA OPERATIVO (SO):** Programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, y se ejecuta en modo privilegiado respecto de los restantes. ColdFusion.

**SISTEMATIZAR:** Organizar, clasificar o reducir a sistema: sistematizaron los datos de los clientes para llevar fácilmente la contabilidad de la empresa.

**SOFTWARE APLICATIVO:** Aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajo. Esto lo diferencia principalmente de otros tipos de programas como los sistemas operativos (que hacen funcionar al ordenador), las utilidades (que realizan tareas de mantenimiento o de uso general), y los lenguajes de programación (con el cual se crean los programas informáticos).

**SOCKETS:** Designa un concepto abstracto por el cual dos programas (posiblemente situados en computadoras distintas) pueden intercambiar cualquier flujo de datos, generalmente de manera fiable y ordenada.

**TABLA BASE DE DATOS:** Tipo de modelado de datos, donde se guardan los datos recogidos por un programa. Su estructura general se asemeja a la vista general de un programa de Hoja de cálculo.

**TCP/IP:** Conjunto de protocolos de red en los que se basa Internet y que permiten la transmisión de datos entre computadoras.

**UML:** Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

**VARIABLE:** Una variable es un espacio de memoria reservado para almacenar un valor que corresponde a un tipo de dato soportado por el lenguaje de programación. Una variable es representada y usada a través de una etiqueta (un nombre) que le asigna un programador o que ya viene predefinida.

## RESUMEN

El proyecto **“análisis y diseño del software para el control de asistencia de estudiantes STEEL”** tiene como propósito la elaboración de una aplicación web que permite realizar registro de datos de estudiantes, registro de asistencia y pagos ya sea por mensualidad, case única o clase especial de la escuela de capoeira NATIVOS permitiendo con manejo de la información de una manera optima mejorando los procesos administrativos y ayudando a la toma de decisiones.

A lo largo de éste documento, el lector podrá visualizar el proceso que se llevo a cabo para la elaboración del software STEEL, los problemas encontrados durante el estudio realizado, la metodología que se implemento para el desarrollo, el lenguaje de programación que se utilizo y sus componentes para su correcta funcionalidad al igual que los requerimientos de maquina (computador).

## INTRODUCCIÓN

Conforme la humanidad se embarca en éste mundo de constante cambio, se puede dar cuenta que las técnicas utilizadas para desenvolverse en actividades típicas con el pasar del tiempo son obsoletas abrumándolos al observar como la tecnología avanza a pasos gigantescos, siendo la única opción para enfrentarse a este proceso la adaptación a los avances tecnológicos que se encuentran a diario; la implementación de software para el control de inventarios, citas médicas, registro de notas, entre otros han ayudado a una mejor administración y productividad en los distintos sectores empresariales, de manera que las actividades que antes el ser humano desarrollaba con gran esfuerzo hoy, las puede hacer utilizando un mínimo de tiempo y energía .

La escuela de capoeira NATIVOS cuenta hoy en día con distintas sedes distribuidas a nivel nacional como lo son en Medellín, Tunja, Villavicencio, Cartagena entre otros y a nivel internacional se encuentran escuelas en Rio de Janeiro y Sao Pablo (Brasil), contando cada una de ellas con una gran cantidad de alumnos entre los que encontramos personas de diferentes clases sociales, edades, creencias religiosas, políticas etc. Durante el tiempo que la escuela ha estado activa se ha enfrentado varios problemas en la realización de los procesos ocasionando que los administradores busquen soluciones rápidas no muy confiables y deficientes.

Es así como nace la idea de construir un sistema de información que permita ser implementado en diferentes sedes de capoeira en donde se pueda tener un control de los alumnos, pago de mensualidades y asistencia a clases; permitiendo de esta manera tener una mejor administración y a la vez un soporte para la toma de decisiones que permita una sostenibilidad y autosuficiencia de la escuela de capoeira NATIVOS.

# **1. PROBLEMA DE INVESTIGACIÓN.**

## **1.1 TITULO DEL PROYECTO**

Análisis y diseño del software para el control de asistencia de estudiantes. (STEEL).

## **1.2 ORIGEN DEL PROYECTO**

El desarrollo de éste proyecto nace de la necesidad de poder ejercer un control y un apoyo a la escuela de capoeira NATIVOS en los procesos que cotidianamente se llevan a cabo, como lo son el pago de mensualidad, registro de asistencia y datos de los alumnos entre otros. En los ya mencionados procesos se ha evidenciado fallas graves en el seguimiento de éstos, afectando el buen servicio a los estudiantes y la sostenibilidad de la escuela.

Gracias a la implementación del sistema de información STEEL, la escuela tendrá a una herramienta útil que le permitirá estar a la vanguardia y un paso delante de las demás escuelas de capoeira apoyando su administración y la tomas de decisiones.

## **1.3 JUSTIFICACION**

La implementación de una herramienta tecnológica para el desarrollo de los procesos que se llevan a cabo dentro de la escuela de capoeira NATIVOS, permitirá que pueda estar a la vanguardia de los avances tecnológicos como lo afirma FERDINAND DRUCKER, Peter. Sistemas de información en las organizaciones: una alternativa para mejorar la productividad general en las pequeñas y medianas empresas en Revista de ciencias sociales (Ve) No.002 (noviembre 2004); pisen 322-337 diciendo “Las empresas se hacen más competitivas en la medida que logran desplazar a otras en los mercados con procesos innovadores que incluyen nuevas tecnologías y nuevas maneras de hacer las cosas”.

Ésta herramienta también permitan manejar la información de forma segura y eficiente y así tener una fuente de datos verídica para la toma de decisiones y

ofrecer un servicio de alta calidad para los alumnos de la escuela. El sistema de información será una herramienta primordial en la realización de los procesos administrativos cotidianos de la escuela tales como el registro de alumnos, la toma de asistencia de los mismos y el pago de mensualidades (costo de las clases) ya que reducirá los tiempos de ejecución de estas tareas, dando fin con las falencias observadas durante el seguimiento de sus procedimientos.

## **1.4PLANTEAMIENTO DEL PROBLEMA**

### **1.4.1. Descripción del problema**

La escuela de Capoeira Nativos es una empresa que ha estado activa desde hace más de 20 años en Colombia, enseñando el arte marcial brasilero conocido como Capoeira. Durante éste tiempo la inexperiencia y problemáticas sobre los procesos que se realizan a diario han llevado al gerente a sus administradores a buscar soluciones rápidas, las cuales muchas veces no son efectivas.

Para determinar las falencias de los procesos administrativos que se llevan actualmente en la escuela, se realizó un análisis de dichos procesos apoyado en entrevistas realizadas al administrador y otra a uno de los estudiantes; mediante ello se logró evidenciar una constante pérdida de información en éstos procesos, a pesar de múltiples intentos por solucionarlos aún no se ha tenido una resultancia óptima.

Además de lo anteriormente descrito se logró evidenciar las falencias plasmadas a continuación:

1. La escuela no cuenta con una herramienta tecnológica que permita un control y gestión en los procesos que realiza la administración.
2. Se observa la pérdida de información de la venta de artículos deportivos.
3. Pérdida del registro de asistencia de estudiantes a clases.
4. Pérdida de facturas de pago de mensualidad, casos únicos y especiales.
5. Pérdida de Información personal de los alumnos y profesores originando un descuadre de dineros y una deserción de alumnos ocasionado que la escuela cada vez luce por su sostenibilidad.

### **1.4.2. Formulación del problema**

El manejo que se le da a la información de los procesos a solucionar por el software en la escuela de Capoeira Nativos y que ha llevado a una pérdida de la misma es el siguiente:

1. En la asistencia a clase de los alumnos se cuenta con éste mecanismo: imprimen en cartones un calendario del año presente en donde el alumno lo diligencia con sus datos básicos y marca el día en que recibió la clase, sin embargo al no tener una organización en ocasiones los alumnos no diligencian la ficha, ésta se pierde el alumno se la lleva a la escuela, generando un inconsistencia en caja debido a no contar con dichas fichas.
2. En el pago de mensualidad el mecanismo es el siguiente: al momento de realizar el pago de la mensualidad, se genera un recibo de caja del cual se conserva una copia para el cierre de caja, aunque éste sistema es muy común y efectivo, sin embargo no ésta excepto de la pérdida de facturas un problema común en la escuela.
3. En cuanto a la información de los alumnos (datos personales) el mecanismo es el siguiente: se piden los datos básicos de los alumnos escribiéndolos en un archivo de Excel, el cual muchas veces se a perdido o dañado ocasionando una pérdida de información y obligando al administrador a realizar jornadas de actualización de datos, ello para tratar de recuperar ésta información. Al ser frecuentes estas situaciones ocasionan molestia para los alumnos y una pérdida de tiempo para el administrador.

El sistema de información STEEL ayudará al administrador a llevar un control de sus actividades cotidianas, logrando una mejor organización, reduciendo tiempos, ahorrando papelería, logrando tener acceso a datos de manera rápida y confiable tanto de los alumnos como de los pagos y asistencia a clase ayudando a los procesos administrativos siendo una herramienta importante e indispensable para La escuela.

## **1.5OBJETIVOS**

### **1.5.1. Objetivo General**

Diseñar, construir e implementar un sistema de información de registro y control de asistencia de los alumnos para la escuela de Capoeira NATIVOS soportándose en una base de datos para de esta manera facilitar la toma de decisiones y su administración.

### **1.5.2. Objetivos específicos**

1. Recopilar la información necesaria para descubrir falencias en los procesos llevados a cabo por la escuela de Capoeira Nativos.
2. Organizar la información enfocándose en las fallas con más urgencia de una solución tecnológica.
3. Diseñar la estructura para la construcción del software en donde se aclaren sus alcances y limitaciones.
4. Establecer los elementos que se utilizaran en la construcción del software de igual forma los recursos tecnológicos.
5. Construir un modulo en donde se pueda Registrar, visualizar y actualizar los datos de los alumnos de la academia mediante la implementación de una base de datos.
6. Elaborar módulos en donde se pueda registrar y consultar la asistencia a clase, los pagos de mensualidad, clases únicas y especiales.
7. Generar reportes estadísticos del registro de asistencia de los alumnos.
8. Presentar la opción de visualizar la información en formatos de Excel y Word.

## **2. MARCO REFERENCIAL**

### **2.1 MARCO TEÓRICO**

#### **2.1.1. Metodología aplicada Métrica v3**

El software STEEL estará basado en metodología METRICA V3, es una Metodología de Planificación útil, para el Desarrollo y Mantenimiento de Sistemas de información. Promovida por el Ministerio de Administraciones Públicas del gobierno español para la sistematización de actividades del ciclo de vida de los proyectos software, por medio de esta metodología se pueden definir las estrategias que se utilizaran para el logro del objetivo propuesto; de esta manera también permitirá que el producto final pueda satisfacer las necesidades y requerimientos de los usuarios, haciendo uso del estándar ISO/IEC 12207 (InformationTechnology / Software LifeCycleProcess) en donde se incluyen las actividades que son aplicadas desde la definición de los requisitos, la adquisición y configuración del sistema hasta la culminación del mismo, proporcionando de esta manera una estructura adecuada tanto para los usuarios, programadores, desarrolladores entre otros individuos que interactúen con el sistema.

La norma ISO/IEC 15504 SPICE Software ProcessImprovementCapabilityDetermination, (Determinación de la Capacidad de Mejora del Proceso Software) el cual es un modelo para la evaluación y mejora de los diferentes procesos de desarrollo y mantenimiento de los sistemas de información de productos software. Para el desarrollo del software STEEL nos basaremos en etapas claves que posee esta metodología, ya que esta estructurada principalmente en el desarrollo y construcción de proyectos software, las cuales adoptaremos de la siguiente manera:

#### **Plan de sistemas de información.**

Tiene como objetivo la obtención de un marco de referencia para el desarrollo de sistemas de información que responda a los objetivos estratégicos de la organización, se generara una descripción de la situación actual de la escuela en sus diferentes academias con el cual se construirá el punto de partida del software incluyendo un análisis técnico y de riesgos.

- **Análisis de la necesidad del software:** se analiza los procesos que se verán afectados con la implementación del software identificando las necesidades de la escuela elaborando un plan de desarrollo para iniciar con el aplicativo.
- **Identificación de alcances del software:** se determinaran los objetivos estratégicos de la escuela de Capoeira que deben ser considerados en el desarrollo del software.
- **Determinación de responsabilidades:** Se verifica los procesos en los cuales se encuentran las falencias y se solucionaran con el software a desarrollar informando a los usuarios afectados y pidiendo su colaboración para de esta forma lograr alcanzar un alto nivel de satisfacción.

### **Definición y organización.**

Se detalla el alcance del proyecto y se organizara el equipo de personas que lo llevaran a cabo con la ayuda de un calendario de ejecución; gracias a esto se lograra una definición mas detallada del proyecto en cuanto a objetivos, procesos afectados, participantes, resultados y fechas de entrega.

- **Especificación del Ámbito y Alcance:** en esta actividad se define el alcance del software esto con los objetivos específicos, los usuarios a los cuales el aplicativo automatizara los procesos participaran de forma activa en la definición de los objetivos.
- **Organización:** se determina las funciones de los responsables de la dirección y seguimiento del software, Adicionalmente, se concretan aspectos como estándares de documentación.
- **Definición del plan de trabajo:** se determinara todos los módulos finales del software con la fecha de entrega para cada uno de ellos teniendo en cuenta la disponibilidad de los usuarios, se detallan las actividades estimando los tiempos y los responsables de dicha tarea y que resultados se espera.
- **Comunicación del plan de trabajo:** en esta etapa se comunicara a los usuarios la elaboración del software con el objetivo de que se logre una aceptación y conozcan los objetivos que se lograrán y la metodología de trabajo que se implementara.

### **Identificación de requisitos.**

Se especifica requisitos de información de la escuela de Capoeira, para lograr esto se estudiara los procesos que se llevan a cabo a diario en la academia realizando secciones de trabajo con los usuarios, analizando cada proceso tal y como debería ser y no como se viene desarrollando actualmente ya que estos pueden estar influenciados por otro método de trabajo organizándolos según su prioridad.

- **Estudio de los procesos:** Se estudiara cada proceso de la academia incluido el plan de desarrollo del software, se identificarán actividades, la información que afecta una con otra y las unidades organizativas que hacen parte del desarrollo de cada actividad. Para llevar a cabo esto se llevarán sesiones de trabajo con los usuarios implicados en cada uno de los procesos en este caso el administrador de la academia, una vez elaboradas las conclusiones, se elabora el modelo correspondiente a cada proceso, se verifica conexiones entre los distintos modelos con el fin de unificarlos lo mas que sea posible para de esta manera proporcionar una visión global en el contexto de la academia y facilitar una identificación de requisitos mas objetiva.
- **Análisis de las Necesidades de información:** se realiza reuniones continuas en donde se identifica las necesidades de información de cada proceso analizado, elaborando un modelo de información en donde se encontrara las relaciones entre los procesos con el objetivo de tener claro que debe realiza cada proceso, así como la información de entrada y salida de cada uno de ellos.

### **Estudio de los sistemas de información actuales.**

Se realiza un estudio detallado de los procesos actuales de la academia, basándose en la documentación recopilada, teniendo en cuenta la opinión de los usuarios ya que de esta manera se lograra saber su nivel de satisfacción con cada uno de los módulos del software.

- **Alcance y Objetivos de los sistemas de información actuales:** teniendo claro los procesos que se verán afectados por el software se determina que sistemas de información se encuentran dentro de los parámetros del proyecto, se seleccionan los que deben ser analizados, así como el objetivo de estudio de cada uno, para de esta forma establecer la función del software dentro de la academia.
- **Valoración de los sistemas de información:** teniendo claro los procedimientos actuales de la academia, se analizan los problemas reales y

potenciales, se obtiene conclusiones y una valorización, lo mas objetiva posible de cada proceso para influir en la decisión de mejorar el software en los próximos años.

### **Diseño del sistema de información.**

Se define la arquitectura de los sistemas junto con el entorno tecnológico en el cual se va a desarrollar el software con los componentes que harán parte del mismo soportado en actividades de diseño de arquitectura de soporte, el diseño de módulos y el diseño físico de datos.

- Diagramado UML: para desarrollar esta etapa se define la arquitectura general del sistema, en donde se define las partes físicas del software en lo cual se utilizaran herramientas de UML necesarias para al construcción del software como lo son: las clases asociadas, el diseño y la realización de los casos de uso, de la misma manera también se revisa la interfaz de usuario, para el diseño de clases se tiene en cuenta la identificación de las clases, de los tributos y de las operaciones que estos realizan.
- Arquitectura de módulos: se definen los módulos del software y su enlace con los demás, durante su diseño se pueden identificar características o comportamientos relacionados; analizando las interfaces y las acciones de excepción entre los módulos, gracias a esto se obtendrá las normas establecidas para el diseño y las conexiones existentes entre módulos.
- Diseño de datos: para definir la estructura física se utilizara el modelo lógico de datos normalizado, de manera que se tenga presente las características específicas del software, de igual manera se analizaran la ruta de acceso a los datos utilizado por cada modulo; logrando al final de esta tarea garantizar la calidad de las especificaciones del diseño y la viabilidad del mismo obteniendo una aceptación y la construcción del aplicativo.

### **Construcción del sistema de información.**

Se construye el código fuente de cada uno de los componentes junto con los procedimientos de operación y seguridad, se realizan todas las pruebas con el objetivo de asegurar un correcto funcionamiento del sistema que se va a implantar.

- Preparación del entorno de construcción del sistema: se asegurara que todos los medios estén disponibles para llevar a cabo la construcción del software, inicializar la base de datos asegurándose la reservación del espacio de almacenamiento, cargar los datos necesarios, bibliotecas o librerías que se usaran inicializando la base de datos o ficheros, las herramientas generadoras de código, editores, compiladores entre otros.
- Generación de código, componente y procedimientos del sistema: se desarrolla el código de cada una de las fases del software, guiándose por las especificaciones de construcción antes establecidas durante el proceso del diseño, contando con los estándares de calidad.
- Ejecución de pruebas unitarias del sistema: se realizan pruebas de cada uno de los módulos del software, con el fin de asegurarse de su correcto funcionamiento y se ajusta a los requerimientos establecidos para asegurar la calidad del software y su aprobación ante el usuario.
- Ejecución de pruebas de sistema: se realizan la verificación de la correcta funcionalidad del software, tanto la interfaz externa como la interna asegurando que cumpla con lo establecido, comprobando su integración y manejo de la información.
- Definición de usuarios finales: el objetivo es establecer la información requerida que deben tener los usuarios y lograr un uso adecuado del software a implementar, dependiendo del perfil de cada usuario que interactúe con el aplicativo.
- Elaboración de manual de usuario: se elabora un documento dirigido al usuario final en donde se apreciar el uso adecuado del software con sus funciones principales para de esta manera logara una rápida aceptación del sistema por parte del usuario.
- Elaboración de manual técnico: se elabora la documentación pertinente del software solo que mas detallada en donde se pueda observar el ensamblaje del sistema a nivel técnico para su modificación o mejora a futuro.

### **Implementación del sistema de información.**

El objetivo de este proceso es la entrega total y la aceptación por parte de los usuarios del software desarrollado, la realización de las actividades necesarias para la ejecución del mismo.

- Prueba de implementación del sistema: las pruebas realizadas al software abarcan un gran campo empezando por la comprobación de cada detalle de diseño, comprobación de los requerimientos, observar los tiempos de ejecución, verificando de esta manera el correcto funcionamiento del software en el entorno de trabajo.
- Pruebas de aceptación del sistema: En este proceso se realizan las pruebas por parte del usuario, con el objetivo de verificar su funcionalidad y cumplimiento con las tareas establecidas.
- Preparación de mantenimiento para el sistema: el objetivo de esta etapa es la mejora u una versión nueva del software, esto se realiza a partir de las peticiones de mantenimiento en los cuales en la mayoría de los casos se hacen presentes debido a problemas detectados en el mantenimiento guardando en un registro dichos errores para tomar medidas para dar solución a dichos problemas.

## **2.2 MARCO TÉCNICO**

### **2.2.1 Normalización De Base De Datos.**

El proceso de normalización de base de datos consiste en aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional.

- **Para Que Se Normaliza Una Base De Datos.**

La importancia de normalizar una base de datos es para evitar la redundancia de los datos, problemas de actualización de los datos en las tablas y proteger la integridad de los datos.

En el modelo relacional es frecuente llamar tabla a una relación, aunque para que una tabla a sea considerada como una relación tiene que cumplir con algunas restricciones: cada tabla debe tener un nombre único, no puede haber dos filas iguales. No se permiten duplicados y todos los datos en una comuna deben ser del mismo tipo es decir si es numérico debe contener un dato de tipo numérico.

## 2.2.2 PHP

PHP es un lenguaje de programación interpretado (Lenguaje de alto rendimiento), diseñado originalmente para la creación de web dinámicas pero actualmente puede ser utilizado en la creación de otros tipos de programas, las siglas PHP significan (Hypertext Pre-processor). Fue creado originalmente por Rasmus Lerdorf en 1994 y ha evolucionado en el transcurso de los años ahora con su nueva versión PHP 5 la cual está enfocada en la orientación a objetos o programación POO.

- **Características de PHP**

PHP posee varias características que lo hacen indicado para el software que se desarrollara de las cuales se pueden mencionar las siguientes:

- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos y reduce los costos para la empresa que desee una herramienta informática para sus procesos.
- Permite aplicar técnicas de programación orientada a objetos ideales para el software a desarrollar.
- Tiene manejo de excepción

- **Ventajas de usar PHP5**

Algunas de las ventajas de usar PHP como lenguaje de programación son las siguientes:

- PHP corre en (casi) cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en algo así como 25 plataformas, incluyendo diferentes versiones de Unix, Windows 95,98,NT,ME,2000,XP, 7 y Macs. Como en todos los sistemas se utiliza el mismo código base, los scripts pueden ser ejecutados de manera independiente al OS.
- PHP es completamente expandible. Está compuesto de un sistema principal (escrito por Zend), un conjunto de módulos y una variedad de extensiones de código.
- Puede interactuar con muchos motores de bases de datos tales como MySQL, MS SQL, Oracle, Informix, PostgreSQL, y otros muchos. Siempre podrás disponer de ODBC para situaciones que lo requieran.
- Rapidez. PHP generalmente es utilizado como modulo de Apache, lo que lo hace extremadamente veloz. Esta completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.

- **Desventajas de usar PHP5**

- El manejo de errores no es tan sofisticado como ColdFusion o ASP.
- Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado. La ofuscación es una técnica que puede dificultar la lectura del código pero no necesariamente impide que el código sea examinado.
- Debido a que es un lenguaje interpretado, un script en PHP suele funcionar considerablemente más lento que su equivalente en un lenguaje de bajo nivel, sin embargo este inconveniente se puede minimizar con técnicas de cache tanto de en archivos y memoria.
- Las variables al no ser tipadas dificulta a los diferentes IDEs para ofrecer asistencias para el tipeado del código, aunque esto no es realmente un inconveniente del lenguaje en sí. Esto es solventado por Zend Studio añadiendo un comentario con el tipo a la declaración de la variable.

### 2.2.3 MYSQL

MYSQL es un servidor de base de datos relacionales con el cual se administra la misma, desarrollado y proporcionado por MYSQL AB. MYSQL Ab es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de base de datos MYSQL.

- **Características de MYSQL.**

Algunas de las características más importantes que posee MYSQL son las siguientes:

- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando ñame pipes y en sistemas Unix usando ficheros socket Unix.
- Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).

- **Para que sirve MYSQL.**

Con MYSQL podemos realizar búsquedas en nuestras tablas a través de lenguaje SQL, basta con realizar una consulta de las mismas y la información requerida se ubica con gran precisión permitiendo reducir el tiempo de búsqueda de datos para un mejor control de la información.

### 2.2.4 HTML

Siglas de *HyperTextMarkupLanguage* (lenguaje de marcado de hipertexto) es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes

## **2.2.5 XAMPP**

Es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl, El programa está liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris y MacOS X.

- **Características**

XAMPP solamente requiere descargar y ejecutar un archivo Zip, tar o exe, con unas pequeñas configuraciones en alguno de sus componentes que el servidor Web necesitará. XAMPP se actualiza regularmente para incorporar las últimas versiones de Apache/MySQL/PHP y Perl. También incluye otros módulos como OpenSSL y phpMyAdmin. Para instalar XAMPP se requiere solamente una pequeña fracción del tiempo necesario para descargar y configurar los programas por separado.

## **2.3 ANTECEDENTES**

En la actualidad se hay sistemas de información encargados del control de asistencia, siendo utilizados con más frecuencia en el sector empresarial que educativo; sin embargo, no hay uno enfocado en escuelas de aprendizaje deportivo como Taekwondo, Karate, Aikido y en éste caso Capoeira cuyas necesidades aunque muy parecidas cambian en algunos aspectos, ya que no es necesario registrar tiempo de almuerzo, llegadas tarde u horas extras entre otros. El no poder adquirir un sistema de información debido a su alto valor de licenciamiento ocasiona que múltiples escuelas deportivas no actualicen sus métodos de administración llevándolos a un atraso tecnológico constante, en el mercado encontramos los siguientes sistemas encargados del control de asistencia:

### **2.3.1 EasyClocking.**

El EasyClocking software le ayuda a administrar el tiempo y asistencia de sus empleados, de una manera más fácil, más eficiente y asequible. El intuitivo y fácil de usar interfaz reducirá dramáticamente el número de horas requeridas para procesar el tiempo y asistencia de sus empleados, y calculara automáticamente el total de horas trabajadas, incluyendo las horas extras, tiempo ausentes, vacaciones, y días festivos. Con el clic de un botón los datos de los empleados pueden ser importados para crear varios reportes informativos para la nomina de sueldos.

La solución de control de asistencia de EasyClocking, reemplaza los métodos tradicionales de coleccionar el tiempo y asistencia y elimina por completo la recopilación de información de nómina manual además trabaja con un lector de huella digital para registrar la asistencia de los empleados.

### **2.3.2 FSAcceso**

Por medio de este programa se controla el acceso a empresas, edificios, colegios, predios o cualquier otra área restringida comunicándose directamente con equipamiento de control de acceso como pueden ser relojes, molinetes, barreras, utilizando códigos de barras proximidad o biométricas.

Permite la administración de funcionarios, visitantes, prestadores de servicios. Registra información personal completa incluyendo una foto mediana la captura con una simple cámara web.

### **2.3.3 SquareNet**

Gracias a este software podrá administrar toda la información relacionada con el cumplimiento y la asistencia del personal, permite definir los horarios de trabajo, pro grupos, por persona, o incluso diario por persona, permitiendo especificar también su puesto de trabajo o centro de costos diario lo que facilita increíblemente el costeo de los empleados en escenarios complejos.

Posee máxima flexibilidad y orientación industrial para cumplir hasta con el mas complejo esquema de horarios y turnos, soporta formulas en Excel para cálculos definidos por el usuario.

### **2.3.4 Biopoint**

Este sistema permite capturar las marcaciones de inicio y fin de jornada en un punto de venta, una pequeña oficina o en general las instalaciones de una empresa con un máximo de 50 empleados.

Esta aplicación registrará por medio de un lector de huellas digitales el instante en el que el empleado inicia o termina la jornada laboral. El sistema guardará la fecha, hora e identidad de la persona que realiza la marcación, de una manera ágil, amigable y virtualmente inmune a fraudes.

El programa reportará en línea las marcaciones realizadas a la oficina o sede principal de la empresa, en los casos en que se aplique, o generará un reporte en Excel, o un archivo de interface, que puede ser consultado o enviado a la sede principal u oficina de gestión humana, vía email, disquete, CD, memoria flash, Internet, sin embargo el costo del lector de huella es alto y no todos pueden tener acceso a este dispositivo.

### 3. CONSIDERACIÓN DE DISEÑO

#### 3.1 Requerimientos Técnicos

La Academia de Capoeira Nativos cuenta con los equipos necesarios para implementar y hacer uso de forma adecuada del software desarrollado, el cual estará soportado en un dominio adquirido por la escuela. Para el desarrollo y puesta en marcha se ha tenido en cuenta todas las especificaciones y requerimientos expuestos durante las indagaciones, de manera que el cliente va a tener un sistema confiable y seguro, de esta manera la puesta en marcha de éste proyecto tendrá una factibilidad técnica favorable ya que se adaptará sin ningún problema al entorno de trabajo encontrado en la academia.

➤ **Software**

- Sistema operativo Windows XP o 7.
- Office 2007,2010.
- Servidor apache superior a versión 5.0.
- Xampp igual o superior a versión 5.0.
- Gestor de base de datos Mysql versión 5.0 o superiores.
- Navegador web internet Explorer 9 o Mozilla 12.0 (recomendado).

➤ **Hardware**

- Ordenador con:
  - Disco duro de 180gb.
  - Memoria de 1gb.
  - Procesador Intel Pentium IV, AMD Athom 64bits
  - Tarjeta de red 10BASE-T: RJ-45

#### 3.2 Factibilidad operativa

En el desarrollo de éste proyecto se contó con la participación de los docentes de la universidad quienes con sus aportes, correcciones y sugerencias lograron orientar y hacer que este proceso se realizara de la mejor manera; de igual forma en la academia se contó con aportes del administrador y algunos de los alumnos más influyentes, quienes con sus requerimientos moldearon un software a

satisfacción de las necesidades expresadas logrando un correcto desempeño de la herramienta de forma dinámica y eficaz dentro de la academia.

### 3.3 Factibilidad económica

#### ➤ Costo de desarrollo

- **Personal del proyecto**

*Tabla 1. Personal del proyecto.*

Personal	
Asesoría Técnica del proyecto	Docente de la Universidad
Asesoría Teórica del proyecto	Docente de la Universidad
Consulta e investigación del proyecto	Estudiante Graduando
Creación del proyecto	Estudiante Graduando
Desarrollador del software	Estudiante Graduando

- **Costo del tiempo del persona del proyecto**

*Tabla 2. Costo del tiempo del personal del proyecto.*

Personal	Cargo	Horas empleadas	Valor por hora	Valor Total
Pablo Fonseca Docente	Asesor Técnico			
Leonardo Pinzón Docente	Asesor Técnico			
Juan c. Alonso Graduado	Creador-Desarrollador			
Total				

- **Costo de materiales del proyecto**

*Tabla 3. Costo de materiales del proyecto.*

Materiales	Cantidad	Valor Unidad	Total
Fotocopias	150	\$ 100	\$ 15.000
Impresiones	150	\$ 150	\$ 20.000

Internet/hora	15	\$ 1.000	\$ 15.000
CD/DVD	2	\$ 2.000	\$ 4.000
Total			\$ 56.500

- **Costos varios del proyecto**

*Tabla 4. Costos varios del proyecto.*

*\*Todos los costos descritos anteriormente serán asumidos por el desarrollador o alumno graduando. Juan Carlos Alonso Bernate.*

Recurso	Cantidad	Valor Unitario	Valor Total
Energía	70 Horas	\$ 600	\$ 42.000
Equipos de computo	COMPAQ Presario CQ40	\$ 950.000	\$ 950.000
Trasporte	50 Pasajes	\$ 1.450	\$ 72.500
Total			\$ 1.064.500

### 3.4 ALCANCES DEL SISTEMA

El sistema de información que se desarrollará será incluido en la pagina web de la escuela en espera de la obtención del dominio se mantendrá de manera local esto ayudara a realizar pruebas y posteriores actualizaciones, éste sistema apoyara al administrador en el desarrollo de procesos administrativos dentro de la academia apoyando a la toma de decisiones basándose en datos verídicos y confiables, esto gracias a que se podrá realizar el registro de los datos de los alumnos en una base de datos de forma segura, permitiendo realizar búsquedas, obteniendo datos actualizados a demás un mejor control en el registro de asistencia a clase por parte de los alumnos y el pago de la mensualidad, clase única y clase especial digitando su código de identificación como alumno para tener un registro de su presencia a clase logrando visualizar la cantidad de alumnos presentes en una grafica ya sea por día o mes.

Por medio de este software no será posible realizar cuentas de venta de artículos deportivos ni inventario de los mismos, no generara facturas. Los alumnos no podrán contactarse con el administrador por este medio, no podrá visualizar horarios o promociones ya que en la página aparece esta información, solo se

tendrá registro de los datos de los alumnos de la escuela a nivel nacional no internacional, el software será implementado en la sede principal de la escuela para posteriormente lograr implementarlo en otras escuelas de formación deportiva.

Gracias a la implementación de este sistema de información la escuela de capoeira tendrá un control más eficiente de la información administrativa manejada diariamente reduciendo el tiempo de elaboración de informes, cuadros de caja y búsqueda de información de datos personales de los alumnos, asistencia a clase y pago de mensualidad, clase única o especial para de esta manera también apoyar a la toma de decisiones administrativas y la creación de eventos, siendo una herramienta muy útil para la escuela con grandes ventajas.

## 4. DISEÑO DEL SISTEMA

### 4.1 DESCRIPCIÓN DEL SISTEMA

El sistema de información que se construirá será un aplicativo web que podrá ser accedido desde cualquier sitio con una conexión a internet desde cualquier navegador.

Este sistema permitirá tener un control de la asistencia, pago de clases e información de los alumnos de la escuela de capoeira NATIVOS con la ayuda de una base de datos para de esta manera tener una fuente de información confiable y segura.

Permitirá la exportación de la información a archivos de tipo doc., como lo son Excel y Word

### 4.2 ARQUITECTURA DEL SISTEMA

*Tabla 5. Arquitectura física del sistema.*

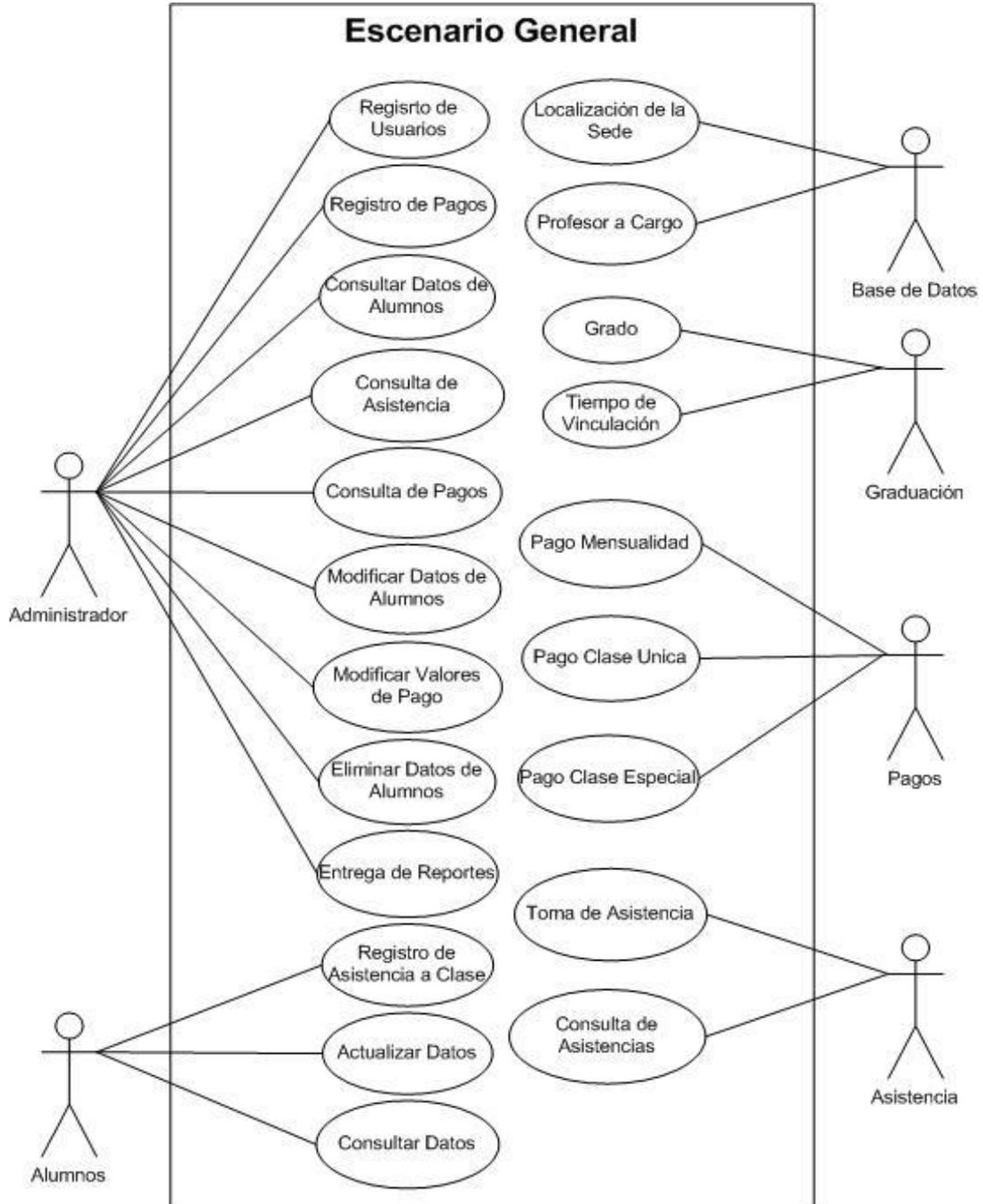
Componente	Tecnología
sistema operativo Cliente	Cualquiera con una conexión a internet
Sistema operador de base de datos	Windows XP, Windows 7
Sistema operativo servidor web	Windows XP, Windows 7
Software para el cliente	Navegador web actualizado
Software para el servidor	Servidor Apache, Xampp, Base de datos MYSQL

### 4.3 MODELO DE CASOS DE USO

Tabla 6. Actores y casos de uso.

Actores	Casos de Uso
Administrador (Usuario)	Registro de usuario
	Registro de pagos
	Consultar datos de alumnos
	Consulta general
	Consulta por alumno
	Consulta de asistencia Por día Por mes
	Consulta de pagos Mes Día
	Consulta de grados
	Consulta de pagos
Alumno (Usuario)	Datos personales
Sedes	Localización de la sede
	Profesor a cargo
Graduación	Grado
Pagos	Pago mensualidad
	Pago clase única
	Pago clases especiales
Asistencia	Toma de asistencia
	Consulta de asistencia

Figura 1. casos de uso escenario general.



### 4.3.1 Descripción general de los Actores

El sistema de información STEEL esta diseñado para el manejo de administradores y usuarios de escuelas deportivas en este caso la escuela de capoeira NATIVOS con el fin de obtener un mejor control de la información manejada en las escuelas como lo son registro de estudiantes, asistencia a clase, pago de mensualidades o clases etc...

El acceso al sistema de información podrá ser accedido desde cualquier sitio con una conexión a internet por medio de un usuario y contraseña, sin embargo el sistema actualmente solo podrá ser accedido por medio local debido a al proceso de adquisición de un domino por parte de la escuela de capoeira.

Gracias a que el aplicativo final estará definido bajo 2 roles, será posible configurar los privilegios según el usuario.

### 4.3.2 Diagramas del modelo de Casos de Uso

Por medio de estos diagramas obtenemos una descripción de las acciones de un sistema desde el punto de vista del usuario gracias a estos diagramas se logra construir un software que puede ser utilizado por la gente en general y no solo por expertos en computación logrando verificar las necesidades que se solucionaran con el sistema de información.

Fig. 2 Caso de Uso. Registro de Usuario



Tabla 7. Caso de uso para el registro del usuario.

Caso de Uso	Registro de Usuarios
Actores	Administrador - Base de Datos
Propósito	Permitir al administrador de la escuela realizar el registro de alumnos que se quieren vincular.
Resumen	El administrador realiza el ingreso de los datos personales de los alumnos, en la base de datos en donde quedara guardada impidiendo la perdida de los datos mas adelante.
Descripción	El usuario administrador ingresa al sistema STEEL con su login y password, cuando son validados los datos podrá acceder a un menú en donde estará la opción de registro y al escogerla le permitirá ingresar los datos básicos del nuevo alumno, también le permitirá en esta opción hacer la asignación de un usuario al nuevo alumno que acabado de ingresar a la base de datos.
Excepciones	Si en el momento del registro el administrador no ingresa los datos en cada uno de los campos obligatorios, el registro del nuevo alumno será denegado.

Fig. 3 Caso de Uso. Ingreso de Usuario



Tabla 8. Casos de uso de ingreso de usuario.

Caso de Uso	Ingreso de usuarios (Alumnos)
Actores	Alumnos - Base de Datos
Propósito	Permitir el ingreso del alumno registrado al sistema.
Resumen	El alumno hace el ingreso al sistema mediante el uso de un login y password que anteriormente se le fue asignado por el administrador en el momento del registro.
Descripción	El estudiante para hacer uso del sistema deberá ingresar el login y password asignados, cuando el sistema lo valide y estos sean correctos, el alumno podrá ingresar y hacer uso de las herramientas que le han sido habilitadas como el de registrar la asistencia a clases, la actualización de datos y la visualización de los pagos realizados, esto le permitirá tener un control sobre la información que ha sido suministrada por el y la que el administrador le ofrece por medio del uso del sistema STEEL.
Excepciones	Cuando el alumno realice el ingreso al sistema los datos ingresados deberán ser correctos de lo contrario el sistema le negara la entrada.

Fig. 4 Caso de Uso. Registro de pagos



Tabla 9. Casos de uso registro de pagos.

Caso de Uso	Registro de Pagos
<b>Actores</b>	Administrador - Base de Datos
<b>Propósito</b>	Permitir al administrador de la escuela realizar el registro de pagos de las clases que los alumnos reciben.
<b>Resumen</b>	El administrador realiza el ingreso de los pagos realizados por los alumnos, en la base de datos en donde quedara guardada impidiendo la perdida de los datos mas adelante.
<b>Descripción</b>	El usuario administrador ingresa al sistema STEEL con su login y password, cuando son validados los datos podrá acceder a un menú en donde estará la opción de registro de pagos, al escogerla le permitirá ingresar los pagos de las clases diferenciando si estas es por pago de mensualidad, pago de clase única o si es una clase especial.
<b>Excepciones</b>	Si en el momento del registro el administrador no ingresa los datos en cada uno de los campos obligatorios, el registro del pago será denegado.

Fig. 5 Caso de Uso. Registro de asistencia a clase.



Tabla 10. Casos de uso asistencia a clase.

Caso de Uso	Registro de Asistencia a Clases
Actores	Alumno- Base de Datos
Propósito	Permitir al alumno registrar la asistencia a clase.
Resumen	El alumno ingresa al sistema y registra su asistencia a clase, en donde se podrá diferenciar si es una clase normal o clase especial.
Descripción	El alumno ingresa al sistema con el login y password cuando son validos podrá realizar la marcación del registro de la asistencia a la clase, de manera que no se generen falencias en el control de la asistencia y se presente mas adelante un inconformismo por alguna de las partes ya sea por el administrador de la academia como de los alumnos, ya que al marcar la asistencia esta se guardara con fecha y hora.
Excepciones	El alumno deberá ingresar de forma correcta los datos para que el sistema no le niegue la entrada y capture el registro o marcación de la clase.

Fig.6 Caso de Uso consulta de datos de alumnos.



Tabla 11. Casos de uso consulta de datos alumnos.

Caso de Uso	Consulta de Datos de Alumnos
<b>Actores</b>	Administrador - Base de Datos
<b>Propósito</b>	Permitir al administrador de la escuela realizar una consulta de los alumnos registrados en el sistema.
<b>Resumen</b>	El administrador iniciara una consulta de los alumnos pertenecen al a escuela de manera esto le permitirá tener un mejor control y actualización de los datos mismos.
<b>Descripción</b>	El usuario administrador ingresa al sistema STEEL, allí encontrara una opción que le permitirá iniciar una consulta, el deberá escoger que tipo de consulta quiere realizar ya sea esta por alumno o en general, cuando ya la haya seleccionado el sistema arrojará la información pedida por administradoras podrá ver los datos todos los alumnos o su defecto de uno de ellos.
<b>Excepciones</b>	En el momento de la consulta el administrador ingrese un código inexistente en el sistema, la consulta se cancelara.

Fig.7 Caso de Uso consulta de datos de datos por alumno.



Tabla 12. Casos de uso consulta de datos por alumnos.

Caso de Uso	Consulta de Datos de Alumnos
<b>Actores</b>	Administrador - Base de Datos

<b>Propósito</b>	Permitir al administrador de la escuela realizar una consulta de los alumnos registrados en el sistema.
<b>Resumen</b>	El administrador iniciara una consulta de los alumnos pertenecen al a escuela de manera esto le permitirá tener un mejor control y actualización de los datos mismos.
<b>Descripción</b>	El usuario administrador ingresa al sistema STEEL, allí encontrara una opción que le permitirá iniciar una consulta, el deberá escoger que tipo de consulta quiere realizar ya sea esta por alumno o en general, cuando ya la haya seleccionado el sistema arrojara la información pedida por administrador así podrá ver los datos todos los alumnos o su defecto de uno de ellos.
<b>Excepciones</b>	En el momento de la consulta el administrador ingrese un código inexistente en el sistema, la consulta se cancelara.

Fig.8 Caso de Uso consulta de asistencia a clase.

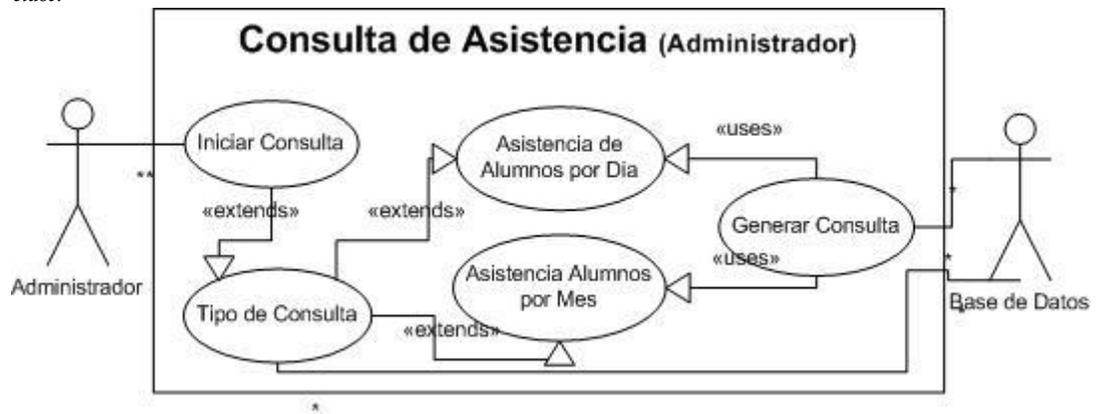


Tabla 13. Casos de uso consulta de asistencia (administrador).

Caso de Uso	Consulta de Asistencia
<b>Actores</b>	Administrador - Base de Datos

<b>Propósito</b>	Permitir al administrador de la escuela revisar el registro de asistencia marcado por alumnos.
<b>Resumen</b>	En el momento de hacer uso del sistema el administrador podrá realizar una consulta de la asistencia por parte de los alumnos.
<b>Descripción</b>	Cuando el administrador ingresa al sistema puede realizar la consulta e asistencia a clase por parte de los alumnos esto con el fin de poder ejercer un control sobre la misma y evitar confusión con la información ya que esta va relacionada con los pagos recibidos y tiene que existir una concordancia tanto en la cantidad de clases como el dinero recibido por las mismas.
<b>Excepciones</b>	Si en el momento del registro el administrador no ingresa los datos en cada uno de los campos obligatorios, la consulta no será generada.

Fig.9 Caso de Uso consulta de pago.



Tabla 14. Casos de uso consulta de pago (administrador).

Caso de Uso	Consulta de Pagos
<b>Actores</b>	Administrador - Base de Datos

<b>Propósito</b>	Permitir al administrador de la escuela revisar el registro de pagos hechos por los alumnos.
<b>Resumen</b>	En el momento de hacer uso del sistema el administrador podrá realizar una consulta de los pagos que los alumnos de la escuela han realizado.
<b>Descripción</b>	Cuando el administrador ingresa al sistema puede realizar la consulta de los pagos de las clases, los cuales son efectuados por los alumnos esto con el fin de poder ejercer un control sobre la mismos y evitar confusión con la información ya que esta va relacionada con los asistencia a clases que los alumnos ha registrado y tiene que existir una concordancia tanto en la cantidad de clases como el dinero recibido por las mismas.
<b>Excepciones</b>	Si en el momento del registro el administrador no ingresa los datos en cada uno de los campos obligatorios, la consulta no será generada.

Fig.10 Caso de Uso consulta de grado.

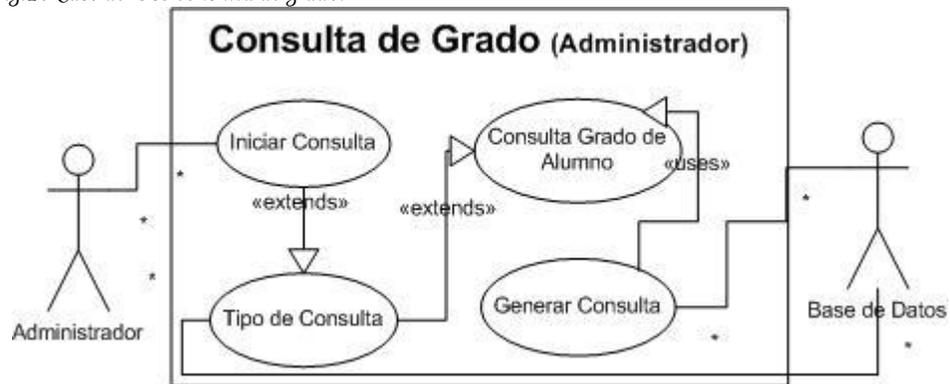


Tabla 15. Casos de uso consulta de grado del alumno (administrador).

Caso de Uso	Consulta de Grado Del Alumno
<b>Actores</b>	Administrador - Base de Datos

<b>Propósito</b>	Permitir al administrador de la escuela revisar que grado porta el alumno.
<b>Resumen</b>	En el momento de hacer uso del sistema el administrador podrá realizar una consulta del grado o corda que posee el alumno.
<b>Descripción</b>	Cuando el administrador ingresa al sistema puede realizar la consulta del grado que el alumno y así poder actualizar la información conforme van subiendo los alumnos en el nivel de preparación.
<b>Excepciones</b>	Si en el momento del registro el administrador no ingresa los datos en cada uno de los campos obligatorios, el sistema no genera la consulta.

Fig.11 Caso de Uso modificar datos.



Tabla 16. Casos de uso modificación de datos (administrador).

Caso de Uso	Modificar Datos
<b>Actores</b>	Administrador - Base de Datos
<b>Propósito</b>	Permitir al administrador realizar modificaciones datos de los alumnos los valores de pago.
<b>Resumen</b>	El administrador podrá realizar modificación en los datos que se tienen de los alumno dentro de la base de datos de igual manera de los valores de los pagos de las clases

<b>Descripción</b>	El usuario administrador ingresa al sistema STEEL con su login y password, cuando son validados los datos podrá acceder a un menú en donde estará la opción modificación podrá elegir ente modificar los datos de los alumnos esto con el fin de evitar inconcordancias de los datos contenidos en la base de datos y la información manejada con verbalmente, una vez realizados los cambios el sistema los guardara y actualizara la información.
<b>Excepciones</b>	Si el administrador no realiza la modificación de forma correcta, bajo las variables establecidas el sistema no guardara los cambios.

Fig.12 Caso de Uso eliminar datos.

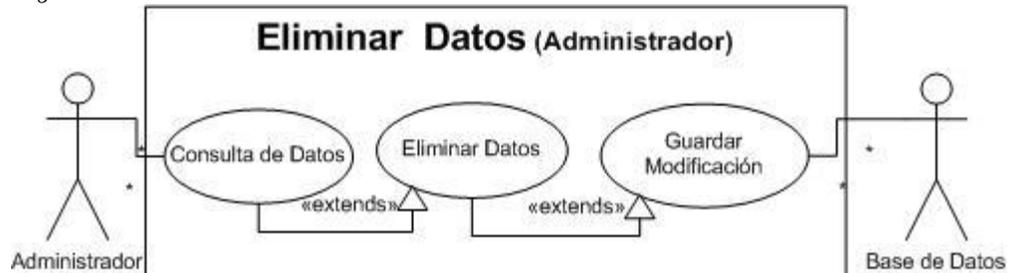


Tabla 17. Casos de uso eliminar datos (administrador).

Caso de Uso	Eliminar Datos
<b>Actores</b>	Administrador - Base de Datos
<b>Propósito</b>	Permitir al administrador eliminar datos de los alumnos.
<b>Resumen</b>	En el caso de que el alumno y no forme parte de la escuela pero figure en el sistema esta información podrá ser eliminada.

<b>Descripción</b>	<p>Cuando el administrador a ingresado al sistema, al elegir la opción de eliminar, primero se realizar la inserción del código a eliminar y el sistema regresara la consulta con los datos que el alumno pero adicionalmente le dará la opción de eliminar por completo los datos del alumno al aceptar y eliminarlos el sistema guardara los cambios y actualizar la información, esto evitar que el exalumno pueda acceder al sistema pues ya no tendrá el usuario que se le había asignado.</p>
<b>Excepciones</b>	<p>Los datos no serán eliminados de forma inmediata sino que no aparecerán activos en el sistema durante cierto tiempo esto con el fin de evitar que el alumno reaparece sus datos ya no existan.</p>

Fig.13 Caso de Uso generar reportes estadísticos.



### 4.3.3 Modelos Objetos

A continuación se muestra el diagrama de clases con sus especificaciones.

Fig.14 Diagrama de clases.

## Diagrama de clases

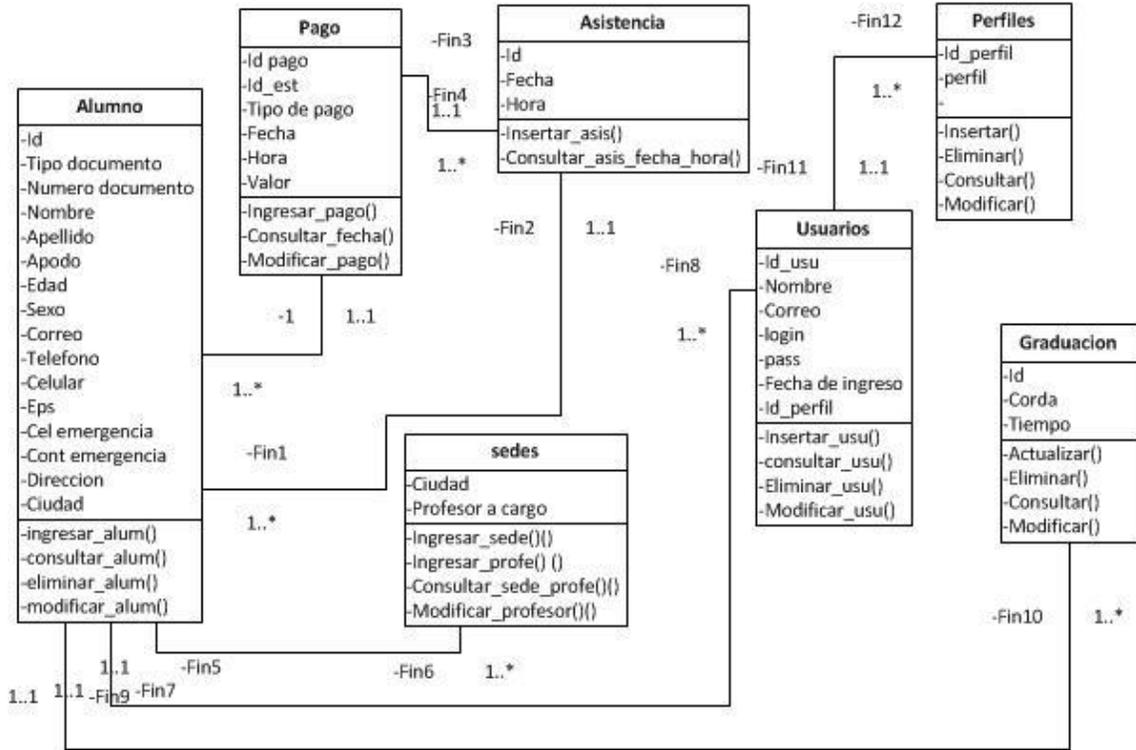


Fig.15 Diagrama de clases, alumno..

<b>Alumno</b>
-Id -Tipo documento -Numero documento -Nombre -Apellido -Apodo -Edad -Sexo -Correo -Telefono -Celular -Eps -Cel emergencia -Cont emergencia -Direccion -Ciudad
-ingresar_alum() -consultar_alum() -eliminar_alum() -modificar_alum()

**Tabla alumno:** esta tabla se encarga de guardar todos los datos personales de los alumnos de la escuela de Capoeira.

Tabla 19. Atributos de la tabla alumnos.

Alumnos			
Nombre atributo	Tipo	longitud	Descripción
Id_alu	Int	4	Identificación único de la tabla
Tipdoc_alu	Varchar	20	Tipo de documento del alumno ya seacedula tarjeta de identidad
Ndoc_alu	Varchar	12	Numero de identificación del documento.
Nomb_alu	Varchar	20	Nombres del alumno
Apel_alu	Varchar	20	Apellidos del alumno.
Capo_alu	Varchar	15	Apodo de Capoeira del alumno si lo tiene.

Edad_alu	Varchar	3	Guarde la edad del alumno
Sexo_alu	Varchar	1	Sexo del alumno
Correo_alu	Varchar	30	Correo del alumno
Tel_alu	Varchar	10	Numero de teléfono fijo del alumno
Cel_alu	Varchar	12	Numero celular del alumno
Eps_alu	Varchar	20	EPS del alumno en caso de emergencia
Contac_alu	Varchar	20	Nombre del contacto de emergencia
Celcont_alu	Varchar	12	Numero de celular del contacto de emergencia.
Dir_alu	Varchar	15	Dirección de residencia del alumno
Ciudad_alu	Varchar	20	Ciudad de origen del alumno

Fig.16 Digrama de clases, pagos.

Pago
-Id pago -Id_est -Tipo de pago -Fecha -Hora -Valor
-Ingresar_pago() -Consultar_fecha() -Modificar_pago()

**Tabla Pago:** donde se guarda la información de pago por parte del alumno ya sea mensualidad, clase única o clase especial.

Tabla20. Atributos de la tabla pagos.

Pagos
-------

Nombre atributo	tipo	contenido
Id_alu	Int	Valor único de la tabla con el id del alumno
Id_pag	Int	Valor único de la tabla con el numero de pago
Tipo_pag	Varchar	Tipode pago ya sea mensualidad, claseúnica o clase especial
Fecha_pag	Datetime	Fecha en que se realizo el pago por parte del alumno
Hora_pag	Datetime	Hora en que se realizo en pago por parte del alumno
Valor_pag	Int	Valor de pago por parte del alumno ya sea por mensualidad, clase única, o especial.

Fig.17 Digrama de clases, asistencia.

Asistencia
-Id -Fecha -Hora
-Insertar_asis() -Consultar_asis_fecha_hora()

**Tabla asistencia:** Esta tabla se encarga de almacenar el id del alumno que se registra como asistente de la clase al igual que la fecha y la hora de su registro.

Tabla 21. Atributos de la tabla asistencia.

Asistencia		
Nombre atributo	Tipo	Contenido
Id_alu	Int	Id del alumno registrado en asistencia a clase.

Fecha_asis	Datetime	Fecha de asistencia a clase.
Hora_asis	Datetime	Hora de registro de asistencia del alumno a clase.

Fig.18 Digrama de clases, usuarios.

Usuarios
-Id -tipo de usuario -Usuario -Contraseña
-Insertar_usu() -consultar_usu() -Eliminar_usu() -Modificar_usu()

**Tabla usuarios:** contiene el id del alumno tipo de usuario para delegar permisos, su usuario y contraseña para el ingreso al sistema de información.

Tabla 22. Atributos de la tabla usuarios.

Usuarios		
Nombre atributo	tipo	contenido
Id_alu	Int	Id del alumno
Tipo_usu	Varchar	Tipo de usuario ya sea administrador o usuario estándar.
Usuarios	Varchar	Usuario de login para el ingreso al aplicativo.
Contra_usu	Varchar	Contraseña de personal de login para el aplicativo.

Fig.19 Digrama de clases, graduación.

Graduacion
-Id -Corda -Tiempo
+Actualizar() +Eliminar() +Consultar() +Modificar()

**Tabla Graduación:** en esta tabla se guarda el color de corda del alumno tiempo que lleva con dicho nivel.

Tabla 23. Atributos de la tabla graduación.

Graduación		
Nombre atributo	tipo	contenido
Id_alu	Int	Valor único id del alumno
Color_grad	Varchar	Color de la corda de Graduación.
Timepo_grad	Varchar	Tiempo que el alumno lleva con un grado.

Fig.20 Digrama de clases, sedes.

sedes
-Ciudad -Profesor a cargo
-Ingresar_sede() -Ingresar_profe() () -Consultar_sede_profe() -Modificar_profesor>()()

**Tabla sedes:** posee la ubicación y el profesor en cargado en cada uno de las sedes a nivel nacional.

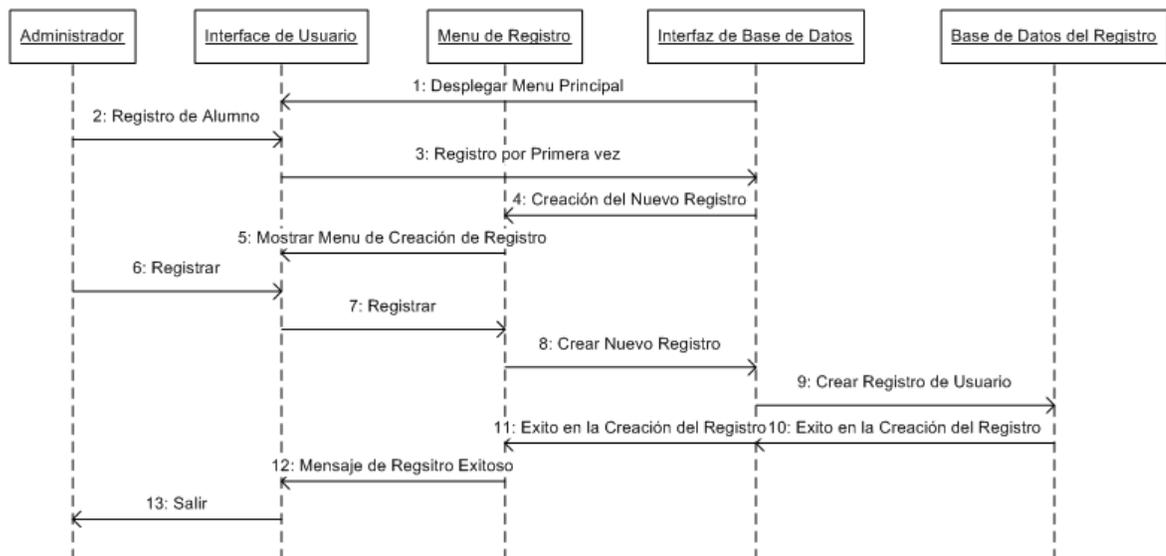
Tabla 24. Atributos de la tabla sedes.

sedes		
Nombre atributo	tipo	contenido
nom_pro	Varchar	Nombre del profesor de cada ciudad
ciudad_pro	Varchar	Ciudad de ubicación de profesor

#### 4.3.4 Modelos Dinámicos.

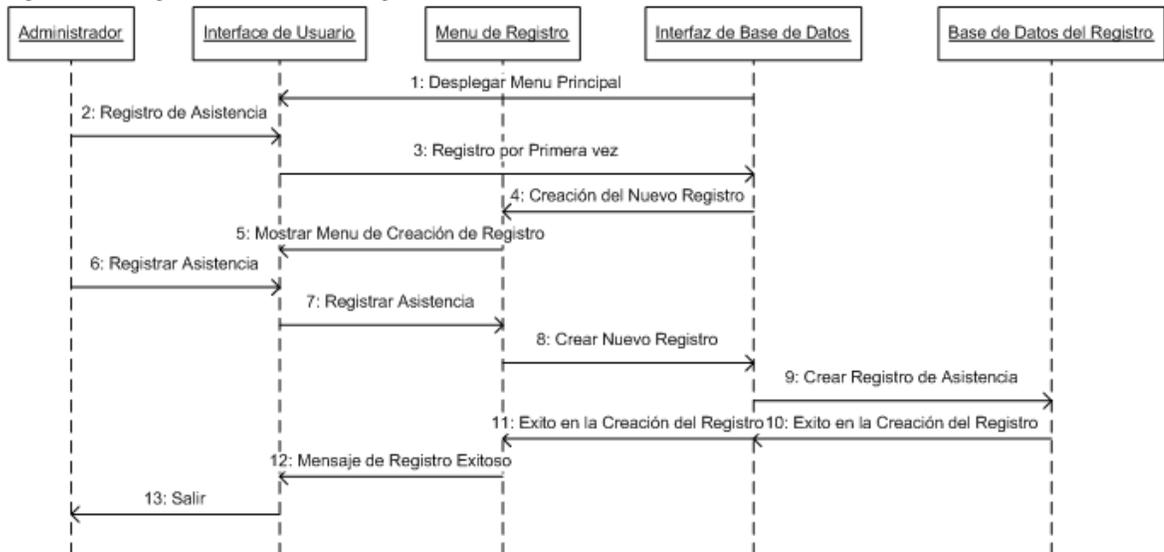
Describe los aspectos que se transforman con el tiempo, éste modelo se utiliza para especificar e implementar las características del funcionamiento del sistema, apoyados en diagramas de secuencia en donde se puede observar el procedimiento de cada caso de uso.

Figura. 21 Diagrama de secuencia, registro de alumno.



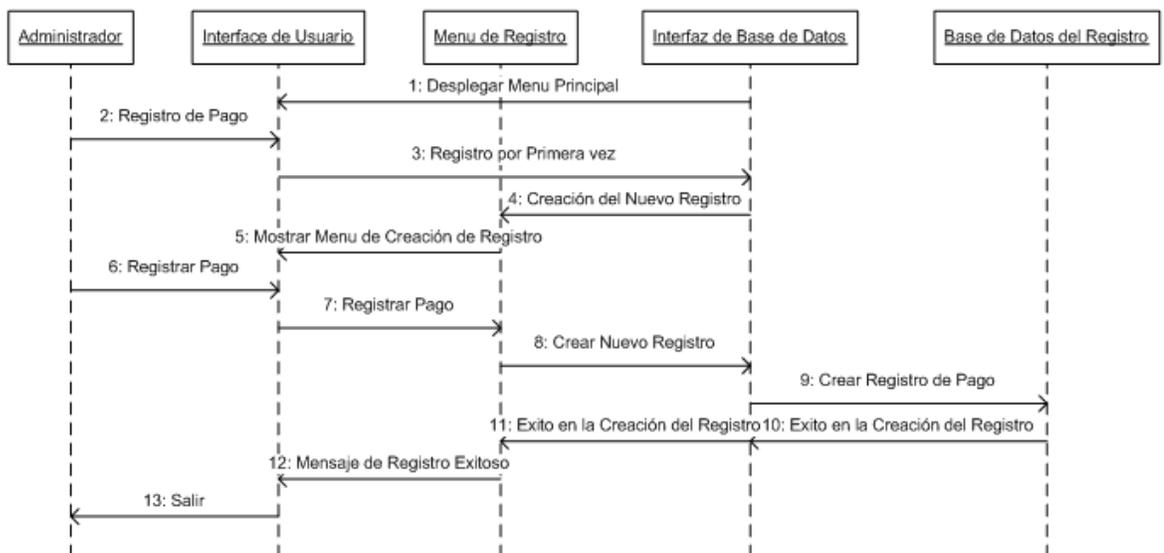
El usuario administrador ingresa los datos de un alumno nuevo en al base de datos por medio de un formulario en donde le pide los datos básicos del estudiante, debe escribir todos los datos requeridos en el formulario para que se puede guardar el registro en la base de datos.

Figura. 22 Diagrama de secuencia, registro asistencia a clase.



El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de registro de asistencia el cual permitira ingresar los datos de asistencia de los alumnos y luego haber realizado el proceso el sistema le devolvera al usuario un mensaje de anuncio que le indicara que el registro de asistencia fue ingresado de manera exitosa, luego podra salir del sistema.

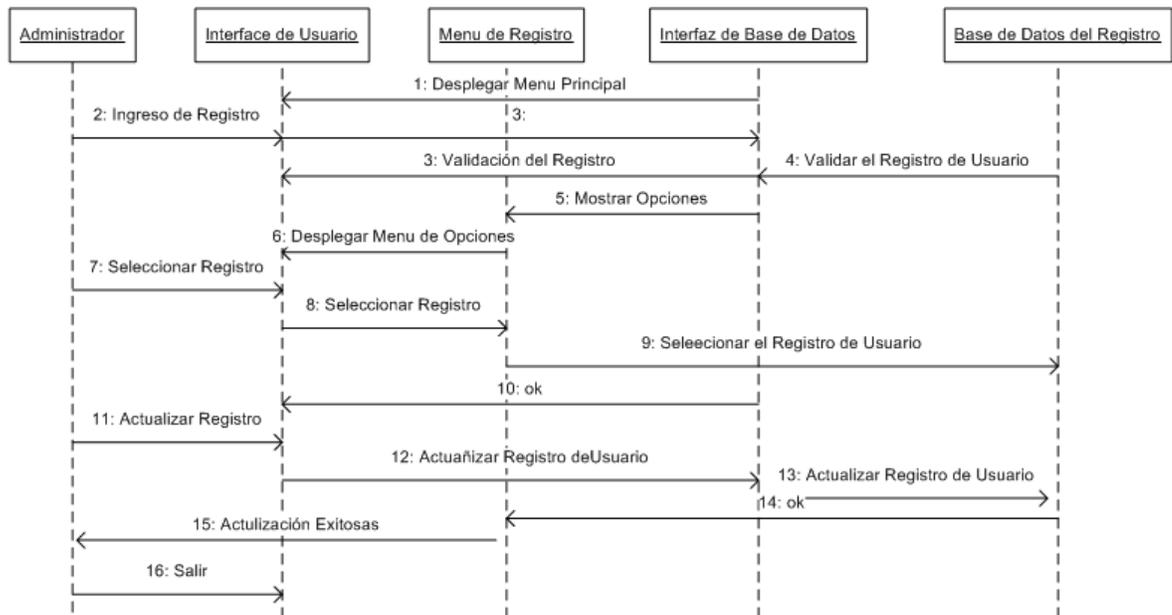
Figura 23 Diagrama de secuencia, registro de pagos de clase.



El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de registro de pagos el cual permitira ingresar el

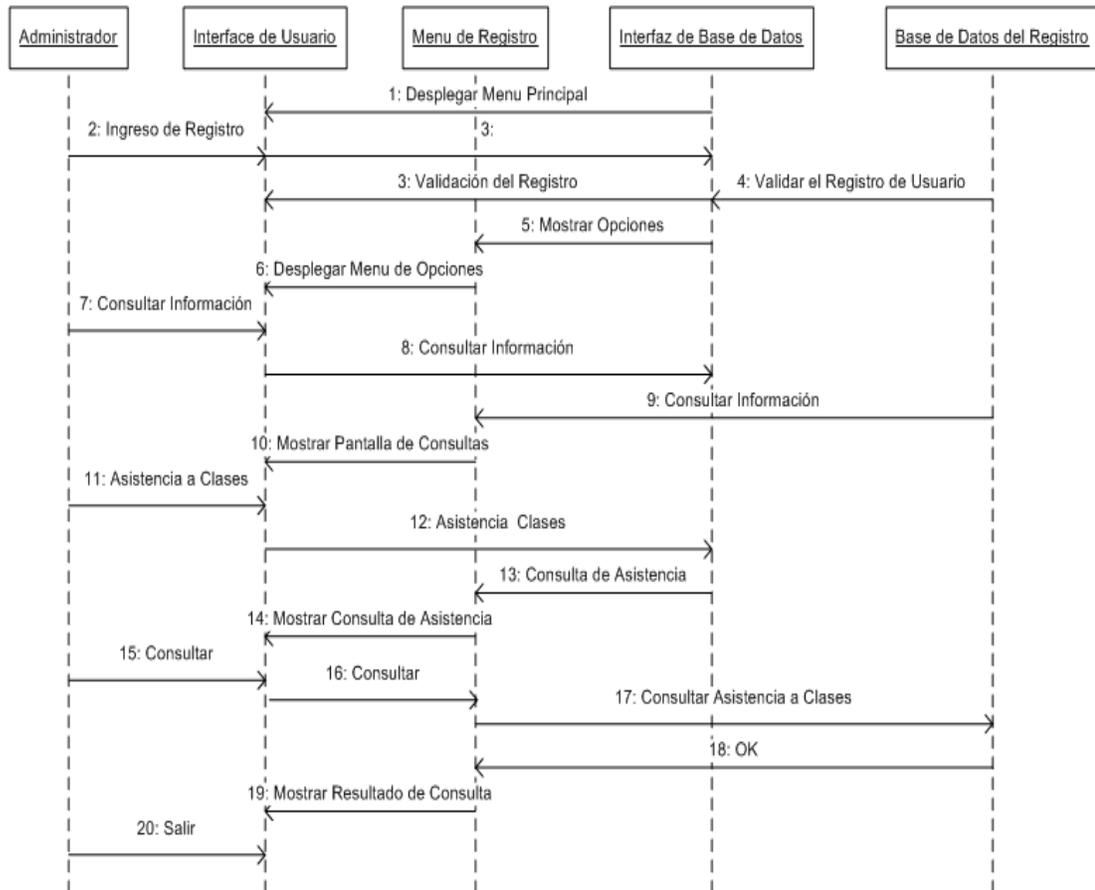
registro de pago de las clases realizados por los alumnos, luego haber realizado el proceso el sistema le devolvera al usuario administrador un mensaje de anuncio que le indicara que el registro de pago fue ingresado de manera exitosa, luego podra salir del sistema.

Figura 24 Diagrama de secuencia, actualizar registro.



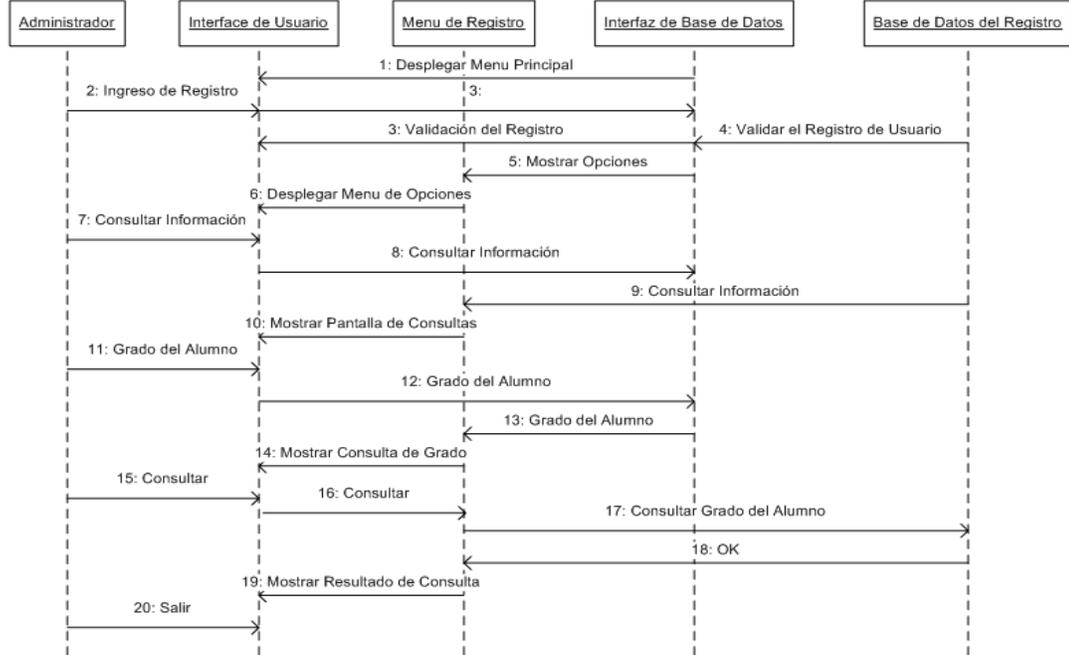
El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de modificacion el cual permitira actualiza el registro de los alumnos, luego haber realizado el proceso el sistema le devolvera al usuario administrador un mensaje de anuncio que le indicara que el registro de alumno fue modificado de manera exitosa, luego podra salir del sistema.

Figura 25 Diagrama de secuencia, consultar asistencia a clases.



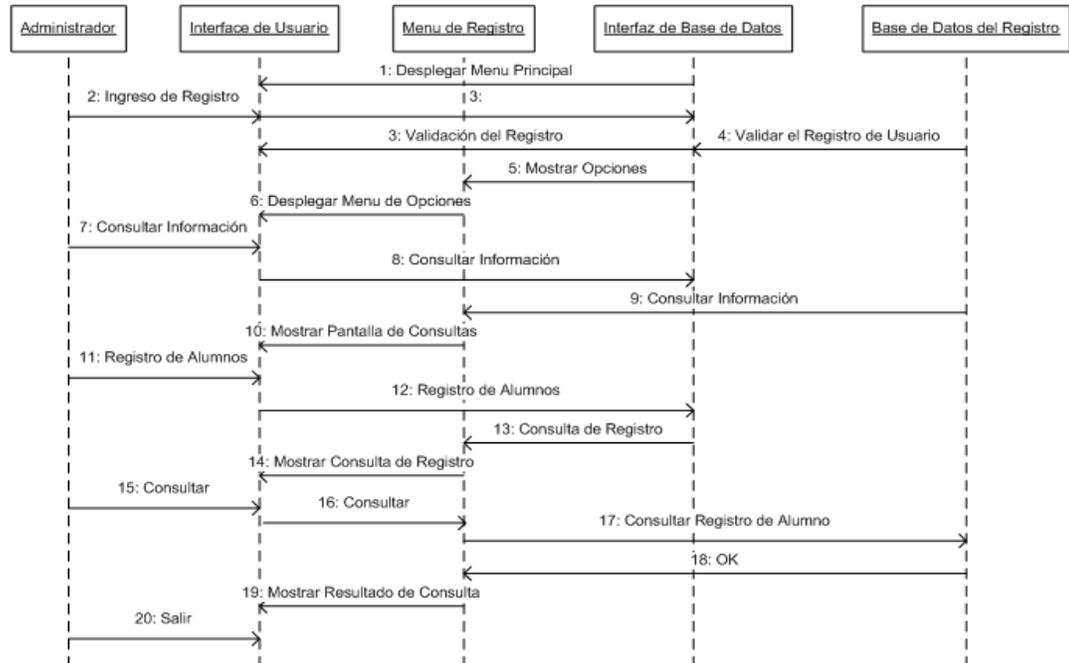
El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de consulta el cual permitira consultar el registro de asistencia a clases de los alumnos, luego haber realizado el proceso de selección de consulta, el sistema le devolvera los datos solicitados, y posteriormente podra salir del sistema.

Figura 26 Diagrama de secuencia, consultar grado de alumnos.



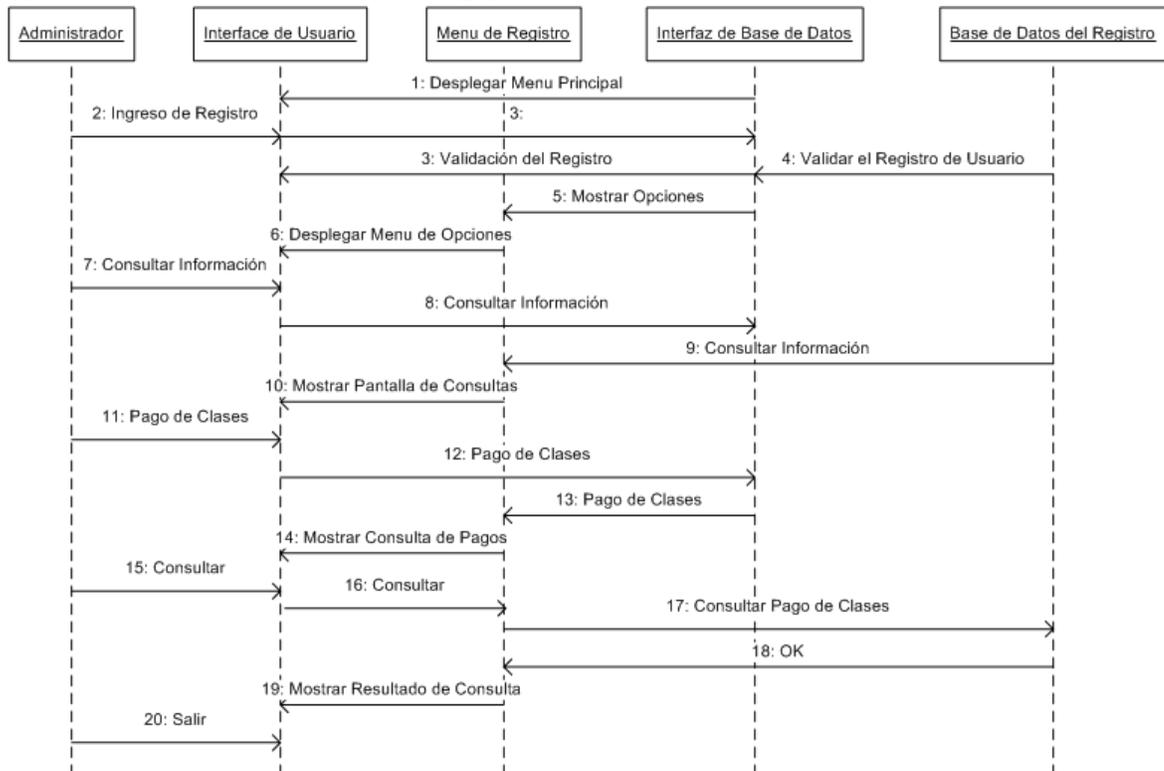
El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de consulta el cual permitira consultar el registro de el grado de los alumnos, luego haber realizado el proceso de selección de consulta el sistema le devolvera los datos solicitados, y posteriormente podra salir del sistema.

Figura 27 Diagrama de secuencia, consultar registro de alumno.



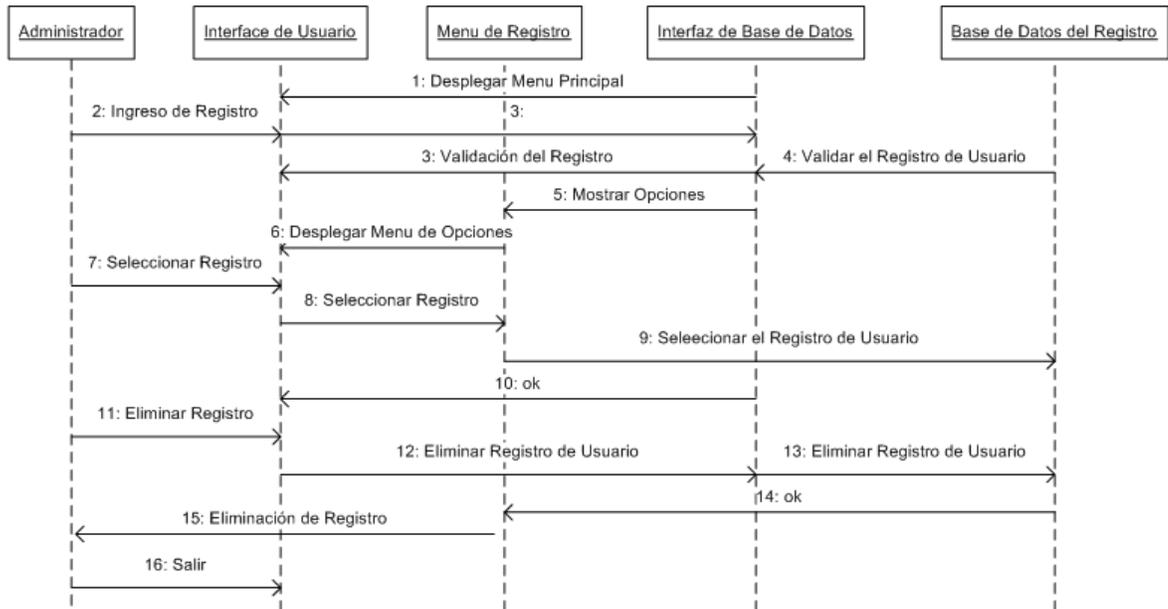
El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de consulta el cual permitira consultar el registro de los alumnos, luego haber realizado el proceso de selección de consulta el sistema le devolverá los datos solicitados, y posteriormente podra salir del sistema.

Figura 28 Diagrama de secuencia, consultar pago de clases.



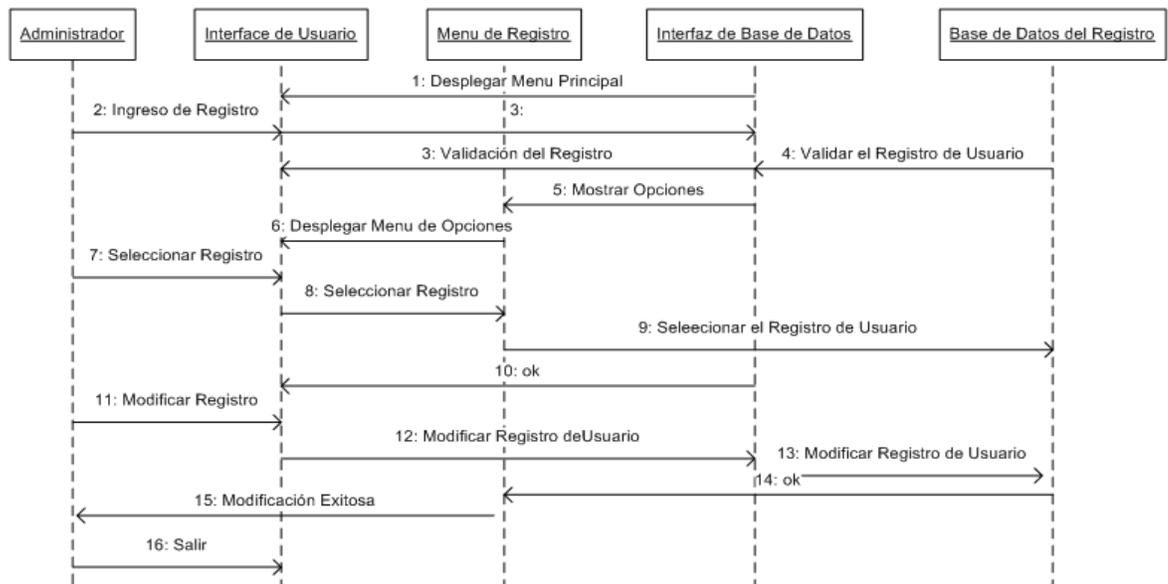
El administrador realiza el ingreso al sistema, este le muestra un menu principal en donde se encuentra el modulo de consulta el cual permitira consultar el registro de pagos, luego haber realizado el proceso de selección de consulta el sistema le devolvera los datos solicitados, y posteriormente podra salir del sistema.

Figura 29 Diagrama de secuencia, eliminar registro de alumno



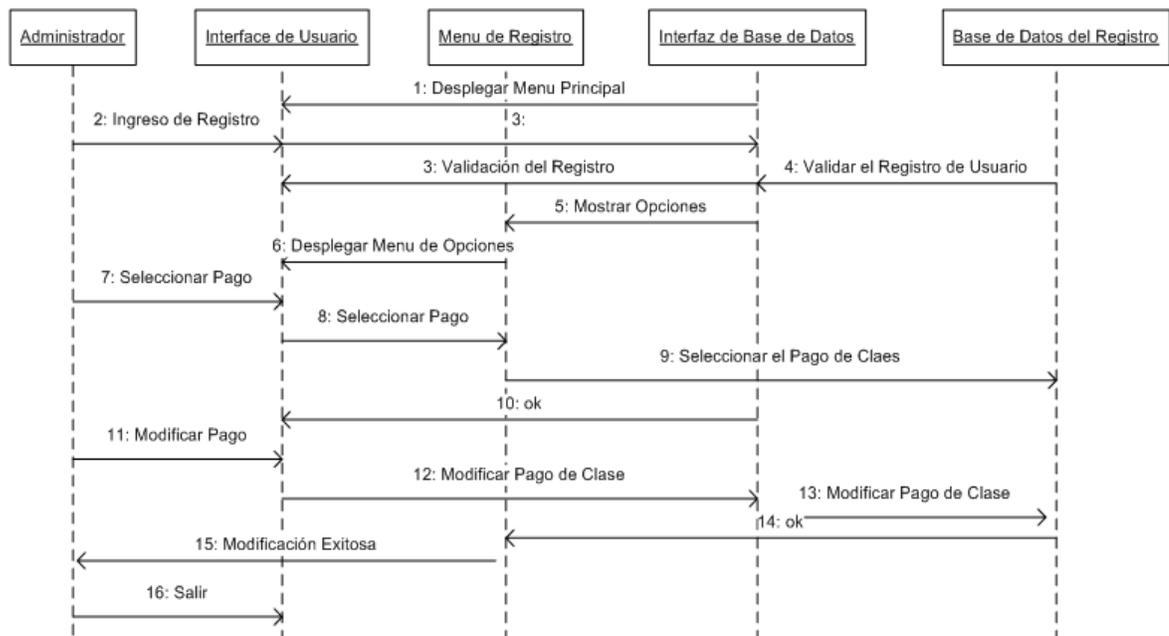
El administrador realiza el ingreso al sistema, este le muestra un menú principal en donde se encuentra el modulo de consulta el cual permitirá consultar el registro del alumno, luego haber realizado el proceso de selección de consulta el sistema le devolverá los datos solicitados y podrá escoger la opción de eliminar, posteriormente la confirmara y así el sistema eliminará, después podrá salir el sistema.

Figura 30 Diagrama de secuencia, modificar registro de alumno.



El administrador realiza el ingreso al sistema, este le muestra un menú principal en donde se encuentra el modulo de consulta el cual permitirá consultar el registro del alumno, luego haber realizado el proceso de selección de consulta el sistema le devolverá los datos solicitados y podrá escoger la opción de modificar el registro de datos de los alumnos, el sistema guardara los cambios, después el usuario podrá salir el sistema.

Figura 31 Diagrama de secuencia, modificar valor de pago de clase.



El administrador realiza el ingreso al sistema, este le muestra un menú principal en donde se encuentra el modulo de consulta el cual permitirá consultar el registró de valor de pago de las clases, luego haber realizado el proceso de selección de consulta el sistema le devolverá los datos solicitados y podrá escoger la opción de modificar el registro de pagos, el sistema guardara los cambios, después el usuario podrá salir el sistema.

## **5. CONCLUSIONES**

Con el desarrollo e implementación del software para el control de asistencia de alumnos en la escuela de capoeira NATIVOS, se logro dar solución a las diferentes problemáticas detectadas durante el proceso de análisis tales como la toma de asistencia, el recaudo de dinero del costo de las clases que se dictan en la academia como también una base de datos con la información básica de los alumnos, permitiendo así contar con datos verídicos que faciliten la toma de decisiones para una mejor administración y prestación de servicios a los alumnos.

En cuanto al entorno estudiantil se puede destacar la implementación de conocimientos adquiridos durante todo el proceso de formación que permitió la aplicación de los mismos dando como resultado un producto final que preste un servicio a un sector de la población solucionando problemáticas expuestas, de la misma manera permitió incursionar el ámbito laboral evaluando las capacidades para enfrentar situaciones reales de la vida cotidiana.

## 6. RECOMENDACIONES Y PROPUESTAS A MEJORAS FUTURAS

Antes de empezar la utilización del sistema STEEL es importante que sea leído el manual de usuario para su buen funcionamiento adicional a esto que los componentes requeridos por el fabricante se cumplan a cabalidad con el fin de evitar inconvenientes.

A modo de propuesta a futuro, se considera que con el objetivo de tener una herramienta más completa y eficiente se considere el desarrollo y mejora de los siguientes módulos como lo son:

- A nivel de inventario, incorporar un modulo en el cual se tenga registro de las mercancías existentes en la escuela a la venta (artículos deportivo y musicales) para el control de entrada y salida de los mismos.
- A nivel de pagos, una mejora que permita la elaboración e impresión de facturas de pago y sea posible realizar búsquedas por número de factura.
- A nivel de registro de asistencia, remplazar el método de id de alumnos para el registro de asistencia y remplazarlo con un lector de código de barras para la optimización de éste modulo.

## BIBLIOGRAFIA

- [1] Weitzenfeld, A. (2005). Ingeniería de software orientada a objetos con Uml, Java e Internet. Cengage Learning Editores: Thomson.
- [2] Patiño, G. (2010). Citas y referencias bibliográficas. Universidad Javeriana, Bogotá. CTP e impresión.
- [3] Flórez, R. (2005). Algoritmos, estructura de datos y programación orientada a objetos. Ecoe Ediciones, Bogotá.
- [4] Cillero, M. (2010). Manuel.Cillero.es: Procesos principales de la Métrica v3 [En Línea].  
Disponibile en:  
<http://manuel.cillero.es/doc/metrica-3/procesos-principales/psi>
- [5] CBASQA. (2008). Desarrollo de Software, SQA, Testing, Servicios Informáticos, Project Management: Metodología Métrica v3-Técnicas y Prácticas. [En Línea].  
Disponibile en:  
<http://cbasqa.wordpress.com/2008/01/10/metodologia-metrica-v3-tecnicas-y-practicas/>
- [6] Informática. (2011). Trabajos investigativos sobre sistemas, software y tecnología informática: Programación orientada a objetos (POO). [En Línea]  
Disponibile en:  
<http://indira-informatica.blogspot.com/2007/09/qu-es-mysql.html>
- [7] Slideshare. (2010). Trabajos Investigativos: PHP 5 y MySQL. [En Línea].  
Disponibile en:  
<http://www.slideshare.net/flaiwebnected/iniciacin-php-5-php-y-mysql>
- [8] Clocking. E. (2011). Easy Clocking Colombia: Sistemas de control de asistencia. [En línea]  
Disponibile en:  
<http://www.easyclocking.com/colombia/control-de-personal.html>
- [9] TIC. (2012). Guía de soluciones: Software, Hardware, Redes y Comunicaciones. [En línea].  
Disponibile en:  
<http://www.guiadesolucionestic.com/sistemas-de-informacion/gestion-de-recursos-humanos/control-de-tiempos-y-asistencias-de-personal/529-biopoint>
- [10] Prieto A, Marlen M. (2004). Internet, Revista de Ciencias Sociales: Sistemas de información de las organizaciones. [En línea].  
Disponibile en:  
<http://redalyc.uaemex.mx/pdf/280/28010209.pdf>