



INSTITUTO TECNOLÓGICO PARTICULAR “LOS ANDES”

TECNOLOGÍA EN ANÁLISIS DE SISTEMAS.

TEMA:

**“ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA
ACADÉMICO PARA EL CENTRO EDUCATIVO DE DESARROLLO
INFANTIL BILINGÜE ANGELITOS DE LUZ, UTILIZANDO EL
LENGUAJE DE PROGRAMACIÓN VISUAL FOX PRO 8.0”**

**Tesis previa a la obtención del Título de Tecnólogo
En Análisis de Sistemas.**

Autor:

Juan Francisco González Pinzón.

Director:

Lic. Steve Mora.

Loja – Ecuador.

2004

Este trabajo lo dedico con gratitud a mi madre, hermanos y en especial a la memoria de mi padre, ya que sin el apoyo de ellos no hubiera sido posible culminar este trabajo de tesis..., por haber merecido de ellos el estímulo, para continuar hacia delante su sabia dirección espiritual retribuyéndoles con este mínimo esfuerzo el sacrificio y la satisfacción del deber cumplido. Ya que ellos son el motivo permanente. A mis verdaderos amigos, amigos compañeros de aula y compañeros andesinos.

Y extra gracias a las personas que me odian, ya que ellos son la gasolina que entra en mi para continuar adelante, por favor no dejen de odiarme por que ustedes son la inspiración para continuar adelante, por favor no dejen de odiarme por que ustedes son la inspiración gracias de todo corazón.

Si usted tiene alguna duda sobre el presente trabajo o requiere información sobre alguna parte de la misma no olvide enviar un e~mail a: foxpanchovisual@yahoo.es, jfpanchogp@gmail.com

Índice.

Portada	i
Página de aprobación	ii
Dedicatoria	iii
Agradecimiento	iv
Introducción	vi

CAPITULO I: Centro de Desarrollo Infantil “Angelitos de Luz.”

- 1.1 Historia.
- 1.2 Servicios que presta.

CAPITULO II: Metodología Orientada a Objetos.

- 2.1 Visual Fox Pro 8.0
 - 2.1.1 MSDN Library.
- 2.2 UML.
 - 2.2.1 Que es UML.
 - 2.2.2 Diagramas de Casos de Uso.
 - 2.2.3 Diagramas de Clases.
 - 2.2.3.1 Atributos y Métodos.
 - 2.2.3.1.1 Atributos.
 - 2.2.3.1.2 Métodos.
 - 2.2.4 Diagramas de Objetos.
 - 2.2.5 Diagramas de Interacción.

2.2.5.1 Diagramas de Secuencia.

2.2.5.2 Diagramas de Colaboración.

2.2.6 Diagramas de Estado.

2.2.7 Diagramas de actividades.

2.2.8 Actores.

2.2.8.1 Definición.

2.2.8.2 Tipos de actores.

2.2.8.2.1 Primarios.

2.2.8.2.2 Secundarios.

CAPITULO III: Análisis de los requerimientos.

3.1 Determinación de los requerimientos.

3.2 Modelo del dominio del sistema.

3.3 Construcción de prototipos.

3.4 Definición de interfaces.

3.5 Identificar casos de uso y esquematizar diagramas

de casos de uso.

CAPITULO IV: Análisis y Diseño Preliminar.

4.1 Narración de los casos de uso encontrados.

4.2 Análisis de Robustez del sistema.

4.2.1 Identificación de los objetos que intervienen

en los escenarios.

4.2.2 Actualización del modelo del dominio.

4.3 Construcción de la Base de Datos, describiendo

cada uno de sus campos.

CAPITULO V: Diseño del sistema.

5.1 Asignación de los comportamientos de las clases

encontradas.

5.2 Construcción del diagrama de secuencia.

CAPITULO VI: Implementación del sistema.

6.1 Escribir y generar código del sistema.

6.2 Ejecutar pruebas de unidad e integración del sistema.

CONCLUSIONES Y RECOMENDACIONES.

BIBLIOGRAFÍA.

INTRODUCCIÓN

Aspiro Con este trabajo teórico practico implementar un sistema informático en el **Centro Educativo de Desarrollo Infantil Bilingüe Angelitos de Luz**, y contribuir al surgimiento de nuevas perspectivas en los estudiantes de informática, en el centro de desarrollo infantil y en otros similares en la utilización de software a la medida para ello me valdré de los conocimientos adquiridos en el Tecnológico “Los Andes”, mi autoaprendizaje y las más modernas técnicas y herramientas para el desarrollo de sistemas informáticos.

Me he planteado algunos objetivos para el desarrollo de este trabajo siendo el principal: realizar el análisis, diseño e implementación de un sistema académico para el **CENTRO EDUCATIVO DE DESARROLLO INFANTIL BILINGÜE ANGELITOS DE LUZ**; con la realización de mi tesis aspiro: dejar al servicio de quienes tengan interés sobre esta temática, un importante instrumento de consulta sobre análisis, diseño e implementación de Sistemas Académicos.

CAPITULO I:

CENTRO EDUCATIVO

DE DESARROLLO

INFANTIL BILINGÜE

“ANGELITOS DE LUZ”

1.1 **HISTORIA.**

El Centro de Desarrollo Infantil Bilingüe Angelitos de Luz fue creado el 6 de Septiembre del 2001, cumple con la misión de proporcionar los conocimientos básicos que necesitan los niños para empezar con su vida estudiantil.

Dados los permanentes cambio sociales, económicos, políticos, educativos, etc., que viene experimentando nuestra sociedad ecuatoriana desde hace algunas décadas, cambios que se vienen expresando en una dinámica social bien diferenciada, donde las diferencias institucionales progresistas no están cumpliendo con el rol para las que fueron creadas, es decir buscar y dirigir esfuerzos y estrategias que procuren encontrar una sociedad mas justa y renovada, que surja de aquella realidad que aun no encontramos y se proyecte con una visión futurista.

EL CENTRO EDUCATIVO DE DESARROLLO INFANTIL BILINGÜE ANGELITOS DE LUZ es un centro de formación integral para el adelanto de los niños, en las áreas Maternal, Estimulación Temprana, Pre – Básica, Primer Año de Básica, Segundo Año de Básica, este Centro por estar en sus inicios no cuenta con un sistema académico. Para contribuir con a las necesidades del CENTRO EDUCATIVO DE DESARROLLO INFANTIL BILINGÜE ANGELITOS DE LUZ, y a las aspiraciones del personal administrativo; considero que es muy necesario la implementación de un software, ya que el personal de oficina realizan sus actividades de forma manual.

1.2 **SERVICIOS QUE PRESTA.**

Educación Inicial.

Área Motriz.

- Desarrollo de actividades motrices:
 - Trepar, saltar, balancearse, rodar, mover brazos y pernas.
- Desarrollar marchas.
 - Caminar en talones y puntas de pie.
- Subir y bajar las escaleras.
- Caminar en dos líneas rectas, paralelas a unos 20 cm. sin pisarlas.
- Ejercicios de sostenerse en un solo pie.
- Ejercicio de patear pelotas.
- Ejecución de la acción que oye o que personalmente visualiza y verbaliza.
- Imitación de ciertos movimientos:
Levantar brazos, aplaudir, dar vueltas.

Áreas Afectivo – Social.

- Realizar contactos sociales breves.
- Participación de la compañía de otros niños, con el predominio del juego.
- Dramatizaciones con relación a:
 - Madre – hijo.
 - Maestra – alumno (lograr una noción adecuada del yo).
- Expresiones momentáneas de cariño, simpatía, comprensión, modestia, etc.
- Actividades de humor para divertir a otros.
- Actividades explorando activamente su mundo, acrecentando su curiosidad; en torno a la música, a cambios de rutinas, a objetos personales.
- Juegos representativos e imitativos, haciendo predominancia de su fantasía.

- Evitaciones del juego solitario o paralelo cuando esta en grupo.

Área Médica.

La atención médica hemos decidido implementarla para lograr un adecuado control del estado de salud de los niños que acuden al Centro de Desarrollo Infantil.

Detectar problemas de salud en los niños del Centro de Desarrollo Infantil.

Actividades.

- Apertura de ficha medica.
- Entrevista con padres de familia.
- Examen coproparasitario, examen de sangre, certificado medico.
- Examen clínico mensual.
- Charla a niños.
- Control antropométrico de niños, etc.

Todas estas actividades mencionadas anteriormente se las efectuara en el consultorio y/o laboratorio con el cual se haga un convenio durante el año lectivo.

Atención Psicológica.

El proceso psicológico se llevara a cabo mediante:

1. Técnicas.

- Fichas personales.
- Entrevistas.

- Inventarios.
- Test psicológico para niños.

2. Planes mensuales de trabajo.

3. Procedimientos didácticos.

- Exposiciones.
- Observaciones colectivas e individuales.
- Análisis, evaluación, y clasificación.
- Itinerarios.
- Diálogos.
- Informes respectivos.

4. Terapias en caso de presentarse.

- Terapias de relajación.
- Terapias en grupo (cognitivo – conductual).
- Terapias familiares.

Servicio de Transporte.

El Centro de Desarrollo Infantil, contrata los servicios de una furgoneta Express, esta brindara un servicio de traslado del infante puerta a puerta; el cual tendrá un valor que deberá ser tratado con el dueño del vehículo, únicamente el Centro de Desarrollo Infantil tendrá un convenio con el propietario del vehículo para aminorar el costo.

Servicio de Lunch.

Este constara de un servicio de alimentación para el infante en el horario de 09:35 AM a 10:00 AM de Lunes a Viernes; el cual en esos días constara de

un menú variado, este tendrá un costo adicional y será cancelado por mesadas.

CAPITULO II:
METODOLOGÍA
ORIENTADA A OBJETOS

La tecnología para soportar el desarrollo de aplicaciones intensivas de datos tuvo una evolución de cuatro generaciones, a saber, sistemas de archivos, sistemas de bases de datos jerárquicos, sistemas de bases de datos en red y sistemas de bases de datos relacionales. En todos los casos, la transición de una generación a otra, ha sido motivada por la necesidad de minimizar los costos de desarrollo (que escalan rápidamente), así como los de mantenimiento y mejora de programas de aplicación.

Los sistemas convencionales (relacionales y prerrelacionales) han servido para satisfacer las necesidades de aplicaciones del ambiente para el cual fueron diseñadas, es decir, de procesamiento de datos en negocios, tales como control de inventario, nóminas, cuentas por cobrar, etc. Sin embargo, tan pronto como esta tecnología abandonó los laboratorios de investigación y registró su marca en el mercado, serias limitaciones comenzaron a ser expuestas. Una variedad de aplicaciones comenzó a ser identificada como difícil para implantarse con el uso de sistemas de bases de datos relacionales. Estas nuevas aplicaciones incluyen diseño auxiliado por computadora, ingeniería, ingeniería de software, gestión y administración de procesos (CAD, CAE, CASE y CAM), sistemas basados en conocimiento (sistemas expertos y "shell" para sistemas expertos), sistemas multimedia que manejan imágenes, gráficas, voz y documentos textuales; modelos estadísticos y científicos y análisis de programas, y sistemas de información geográfica, entre otras.

La metodología Orientada a Objetos consiste en:

- Construir el modelo del dominio de aplicación añadiendo detalles de implementación durante el diseño de un sistema.

Fases.

- Análisis.
- Diseño del sistema.
- Diseño de Objetos.

Análisis.

- Analista construye modelo de situación del mundo real mostrando sus propiedades importantes.
- El modelo del análisis es abstracción resumida y precisa de lo que debe hacer el sistema deseado y no la forma en que se hará.
- Los objetos del modelo deberán ser conceptos del dominio de la aplicación y no conceptos de la implementación de la computadora como son las estructuras de datos.

Diseño del sistema

- El diseñador de sistemas toma decisiones de alto nivel acerca de la arquitectura global.
- Durante el diseño, el sistema de destino se organiza en subsistemas basados tanto en la estructura del análisis como en la arquitectura propuesta.
- El diseñador deberá decidir que características de rendimiento hay que optimizar.

Diseño de Objetos

- El diseñador de objetos construye un modelo de diseño basándose en el modelo de análisis que lleven incorporados detalles de implementación.

La metodología Orientada a Objetos emplea tres clases de modelo para describir al sistema, estas son:

1. **Modelo de Objetos:** describe objetos del sistema y sus relaciones.
2. **Modelo Dinámico:** describe interacciones existentes entre objetos del sistema.
3. **Modelo Funcional:** describe transformaciones de datos del sistema.

Modelo de Objetos

- Describe la estructura estática de los objetos del sistema y también sus relaciones.
- Contiene diagramas de objetos, el cual es un grafo:
 - NODOS = Clases de Objetos.
 - ARCOS = Relaciones entre Clases.

Modelo Dinámico.

- Describe los aspectos de un sistema que cambian con el tiempo.
- Se utiliza para especificar e implementar los aspectos de control del sistema.
- Contienen diagramas de estado.
- Diagramas de estado es un grafo.

Modelo Funcional.

- Describe las transformaciones de valores de datos que ocurren dentro del sistema.
- Contiene diagramas de flujos de datos.
- Un diagrama de flujos de datos representa un cálculo y es un grafo.
 - NODOS = Procesos.
 - ARCOS = Flujo de Datos.

Para el desarrollo de mi tesis utilizare el programa Visual Fox Pro 8.0 en el cual se permite trabajar con objetos.

2.1 **Visual Fox Pro 8.0.-** Es un lenguaje de programación que permite la construcción de software para ser implementado en una determinada empresa o entidad.

2.2 **UML.-** El lenguaje de modelamiento Unificado (UML –Unified Modeling Language)

2.2.1 **Que es UML.**

Es un lenguaje grafico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un determinado lenguaje, esquemas de Base de Datos y componentes de software rehusables es un lenguaje que permite modelar, construir y documentar los

elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE. En el proceso de creación de UML han participado, no obstante, otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores.

2.2.2 Diagramas de Casos de Uso.

Para construir el Modelo de Casos de Uso en la fase de Planificación y Especificación de requisitos se siguen los siguientes pasos:

1. Después de listar las funciones del sistema, se definen los límites del sistema y se identifican los actores y los casos de uso.
2. Se escriben todos los casos de uso en el formato de alto nivel. Se categorizan como primarios, secundarios u opcionales.
3. Se dibuja el Diagrama de Casos de Uso.
4. Se relacionan los casos de uso y se ilustran las relaciones en el Diagrama de Casos de Uso (<<extiende>> y <<usa>>).

5. Los casos de uso más críticos, importantes y que conllevan un mayor riesgo, se describen en el formato expandido esencial. Se deja la definición en formato expandido esencial del resto de casos de uso para cuando sean tratados en posteriores ciclos de desarrollo, para no tratar toda la complejidad del problema de una sola vez.
6. Se crean casos de uso reales sólo cuando:
 - a. Descripciones más detalladas ayudan significativamente a incrementar la comprensión del problema.
 - b. El cliente pide que los procesos se describan de esta forma.

2.2.3 Diagramas de Clases.

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

- En donde:
 - **Superior:** Contiene el nombre de la Clase
 - **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).

- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

2.2.3.1 Atributos y Métodos.

2.2.3.1.1 **Atributos.** Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:

- **public (+).** Indica que el atributo será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.
- **private (-).** Indica que el atributo sólo será accesible desde dentro de la clase (sólo sus métodos lo pueden acceder).
- **protected (#).** Indica que el atributo no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de las subclases que se deriven (ver herencia).

2.2.3.1.2 **Métodos.** Los métodos u operaciones de una clase son la forma en como ésta

interactúa con su entorno, éstos pueden tener las características:

- **public (+).** Indica que el método será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.
- **private (-).** Indica que el método sólo será accesible desde dentro de la clase (sólo otros métodos de la clase lo pueden acceder).
- **protected (#).** Indica que el método no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de métodos de las subclases que se deriven.

2.2.4 **Diagramas de Objetos.** Es utilizado para modelar la estructura estática de los objetos en el sistema.

2.2.5 **Diagramas de Interacción.** Un caso de uso se modela para todos los procesos que el sistema debe llevar a cabo. Los procesos se describen dentro del caso de uso por una descripción textual o una secuencia de pasos ejecutados.

2.2.5.1 **Diagramas de Secuencia.** Un diagrama de Secuencia muestra una interacción ordenada según la secuencia temporal de eventos. En particular, muestra los objetos

participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o activaciones a las que se refieren.

2.2.5.2 Diagramas de Colaboración. Un Diagrama de Colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos (en cuanto a la interacción se refiere). A diferencia de los Diagramas de Secuencia, los Diagramas de Colaboración muestran las relaciones entre los roles de los objetos. La secuencia de los mensajes y los flujos de ejecución concurrentes deben determinarse explícitamente mediante números de secuencia.

2.2.6 Diagramas de estado. Un Diagrama de Estados muestra la secuencia de estados por los que pasa un caso de uso o un objeto a lo largo de su vida, indicando qué eventos hacen que se pase de un estado a otro y cuáles son las respuestas y acciones que genera. En cuanto a la representación, un diagrama de estados es un grafo

cuyos nodos son estados y cuyos arcos dirigidos son transiciones etiquetadas con los nombres de los eventos.

Un estado se representa como una caja redondeada con el nombre del estado en su interior. Una transición se representa como una flecha desde el estado origen al estado destino.

La caja de un estado puede tener 1 o 2 compartimentos. En el primer compartimiento parece el nombre del estado. El segundo compartimiento es opcional, y en él pueden aparecer acciones de entrada, de salida y acciones internas.

Una acción de entrada aparece en la forma *entrada/acción_asociada* donde *acción _ asociada* es el nombre de la acción que se realiza al entrar en ese estado. Cada vez que se entra al estado por medio de una transición la acción de entrada se ejecuta.

Una acción de salida aparece en la forma *salida/acción_asociada*. Cada vez que se sale del estado por una transición de salida la acción de salida se ejecuta.

Una acción interna es una acción que se ejecuta cuando se recibe un determinado evento en ese estado, pero que no causa una transición a otro estado. Se indica en la forma *nombre_de_evento/acción_asociada*.

2.2.7 Diagramas de Actividades. Este tipo de diagrama nos sirve fundamentalmente para modelar el flujo de control de actividades.

La idea es generar una especie de diagrama Pert, en el que se puede ver el flujo de actividades que tienen lugar a lo largo del tiempo, así como las tareas concurrentes que pueden realizarse a la vez. El diagrama de

actividades sirve para representar el sistema desde otra perspectiva, y de este modo complementa a los anteriores diagramas vistos.

Gráficamente un diagrama de actividades será un conjunto de arcos y nodos.

Desde un punto de vista conceptual, el diagrama de actividades muestra cómo fluye el control de unas clases a otras con la finalidad de culminar con un flujo de control total que se corresponde con la consecución de un proceso más complejo.

Por este motivo, en un diagrama de actividades aparecerán acciones y actividades correspondientes a distintas clases. Colaborando todas ellas para conseguir un mismo fin.

2.2.8 Actores.

2.2.8.1 Definición. Un actor es una entidad externa al sistema que realiza algún tipo de interacción con el mismo. Se representa mediante una figura humana dibujada con palotes. Esta representación sirve tanto para actores que son personas como para otro tipo de actores (otros sistemas, sensores, etc.).

- Un usuario puede jugar diferentes roles.
- En la realización de un caso de uso pueden intervenir diferentes actores.
- Un actor puede intervenir en varios casos de uso.
- Identificar casos de uso mediante actores y eventos externos.

2.2.8.2 Tipos de Actores.

2.2.8.2.1 Primarios.

Requieren al sistema el cumplimiento de un objetivo.

2.2.8.2.2 Secundarios.

El sistema necesita de ellos para satisfacer un objetivo.

CAPITULO III:
ANÁLISIS DE LOS
REQUERIMIENTOS.

Los requerimientos son una descripción de las necesidades o deseos de un producto. La meta primaria de la fase de los requerimientos es identificar y documentar lo que en realidad se necesita, en una forma que claramente se lo comuniqué al cliente. El reto consiste en definirlos de manera inequívoca, de modo que se detecten los riesgos y no se presenten sorpresas al momento de entregar el producto.

Para la fase de requerimientos se recomienda:

- Panorama general.
- Clientes.
- Metas.
- Funciones del sistema.
- Atributos del sistema.

3.1 Determinación de Requerimientos.

Es la parte más importante dentro del desarrollo de sistemas y/o sitios web dinámicos, para la ejecución del presente trabajo los requerimientos se dividen en tres partes: la primera parte registro de alumnos, representantes y almacenamiento de notas de los alumnos (en diferentes áreas), la segunda servicios especiales que presta la institución, y la tercera parte administrativa.

Registro de alumnos, representantes y almacenamiento de notas de los alumnos.

Ref.	Requerimiento	Categoría.
R.001	Registrar nuevos Alumnos.	Evidente
R.002	Registrar Representantes.	Evidente

R.003	Presentar área al cual se vaya a matricular a un alumno.	Evidente.
R.004	Ingreso de Notas de Alumnos	Evidente
R.005	Registrar datos de profesores.	Evidente.

Servicios especiales que presta la Institución.

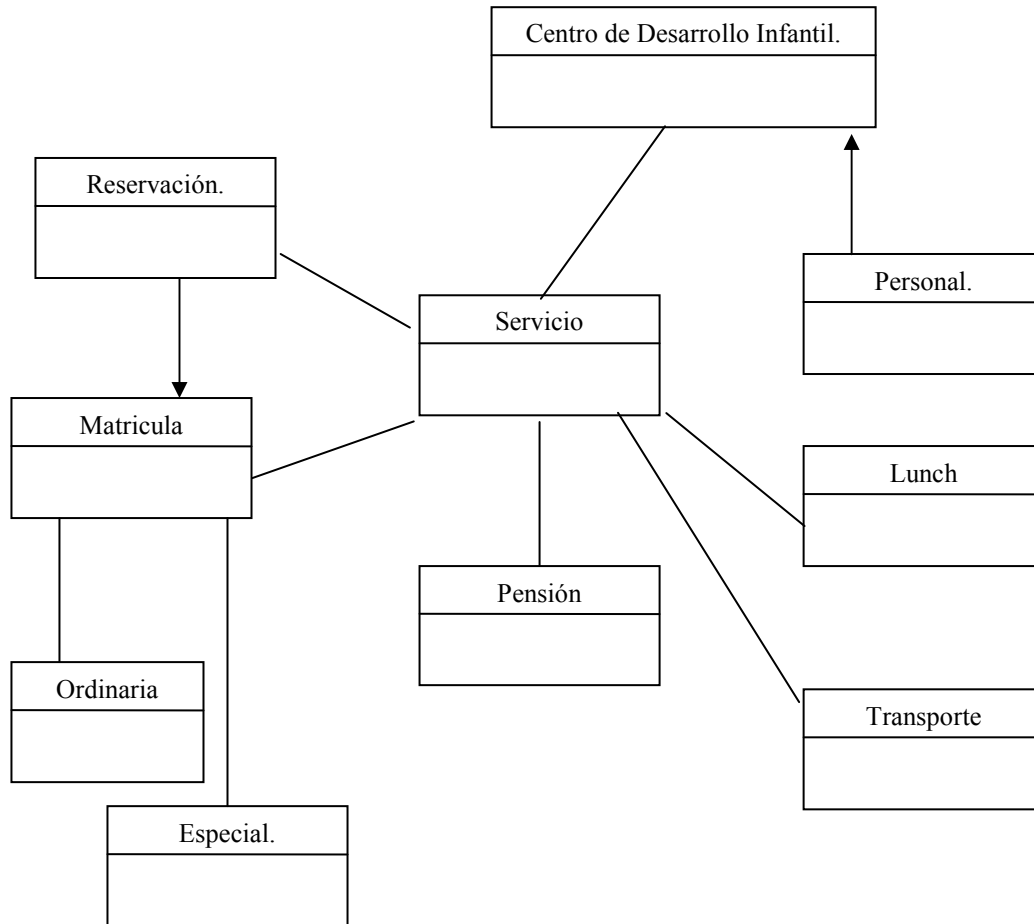
Ref.	Requerimiento	Categoría.
R.006	Servicio de Lunch.	Evidente.
R.007	Servicio de Transporte	Evidente.

Administración.

Ref.	Requerimiento	Categoría.
R.008	Ingreso al sistema mediante un Login y un password.	Evidente.
R.009	Listar los alumnos matriculados por áreas.	Evidente.
R.010	Ingresar los datos generales del representante.	Evidente.
R.011	Listar los datos de los padres dado el nombre del niño.	Evidente.
R.012	Ingresar datos generales del personal.	Evidente.
R.013	Listar los datos del personal que labora en la institución.	Evidente.
R.014	Modificar datos del alumno.	Evidente.
R.015	Modificar datos del personal que labora en la	

	institución.	Evidente.
R.016	Modificar los datos del representante.	Evidente.
R.017	Eliminar datos del personal que labora en la Institución.	Evidente
R.018	Registrar usuarios del Sistema.	Evidente.
R.019	Registrar Asistencia del Alumno	Evidente.
R.020	Registrar Áreas de Estudio	Evidente
R.021	Registro de Materias por área.	Evidente.
R.022	Registrar Matricula.	Evidente.
R.023	Registrar Pensiones.	Evidente.
R.024	Reserva cupo.	Evidente.
R.025	Registrar ficha de ingreso.	Evidente.
R.026	Modificar datos de usuario_ sistema	Evidente.
R.027	Registrar Uniformes.	Evidente.
R.028	Registrar venta de uniformes.	Evidente.
R.029	Modificar información de uniformes	Evidente.

3.2 Modelo del Dominio del Sistema.



3.3 Construcción de Prototipos.

La construcción de los prototipos del sistema se lo define como un modelo a escala de lo real, pero no tan funcional para que equivalga a un producto final, ya que no lleva a cabo la totalidad de las funciones necesarias del sistema final. Proporcionando una retroalimentación temprana por parte de los usuarios acerca del Sistema. Para definir un prototipo siempre se debe establecer cual es su verdadero objetivo, ya que un prototipo puede ser útil en diferentes fases del proyecto, por ello su objetivo debe ser claro. Durante la fase de análisis se usa para obtener los requerimientos del usuario. En la fase de diseño se usa para ayudar a evaluar muchos

aspectos de la implementación seleccionada. El propósito de un prototipo en la fase de Análisis de un proyecto, su principal propósito es obtener y validar los requerimientos esenciales, manteniendo abiertas, las opciones de implementación. Esto implica que se debe tomar los comentarios de los usuarios, pero debemos regresar a sus objetivos para no perder la atención. En la fase de Diseño, su propósito, basándose en los requerimientos previamente obtenidos, es mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una retroalimentación que nos permite mejorar el Diseño de Interfaz. El proceso de desarrollo y empleo de propósito tiene las siguientes características:

- El prototipo es una aplicación que funciona.
- Los prototipos se crean con rapidez.
- Los prototipos evolucionan a través de un proceso iterativo.
- Los prototipos tienen un costo bajo de desarrollo.

3.4 **Definición de Interfaces.**

Se puede definir como componentes se especifican por su interface o interfaces, sin ofrecer conocimiento del diseño e implementación internas del componente. El desarrollo basado en componentes es el proceso de ensamblar la combinación correcta de componentes en la configuración correcta para llevar acabo la funcionalidad deseada para un sistema.

La importancia de definir su objetivo es que en la fase de Análisis de un proyecto, su principal propósito es obtener y validar los requerimientos esenciales, manteniendo abiertas, las opciones de implementación. Esto

implica que se debe tomar los comentarios de los usuarios, pero debemos regresar a sus objetivos para no perder la atención.

En la fase de Diseño, su propósito, basándose en los requerimientos previamente obtenidos, es mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una retroalimentación que nos permite mejorar el Diseño de Interfaz.

Para desarrollar las interfaces se considera los siguientes factores.

- Problemas no estructurados, novedosos y complejos, de información personalizada del usuario, ya que sus salidas no son predecibles y definidas
- Problemas de ambiente Inestable, el profesional también debe evaluar el contexto del sistema.
- Experiencia en diseños similares.
- No se conocen los requerimientos, la naturaleza del sistema es tal que existe poca información con respecto a las características que debe tener el nuevo sistema para satisfacer las necesidades del usuario.
- Los requerimientos deben evaluarse, se conocen los requerimientos aparentes de información pero es necesario verificarlos y evaluarlos.
- Costos altos, donde la inversión involucra gran cantidad de recursos financieros y humanos.
- Altos riesgos, la evaluación inexacta de los requerimientos o el desarrollo incorrecto ponen en peligro a la organización.
- El usuario, donde no está dispuesta a examinar modelos en papel, o no sabe lo que quiere pero lo reconocerá cuando lo vea.

- Tecnologías Nuevas, la falta de experiencia en el uso de dichas tecnologías, junto con el deseo de instalar nuevas tecnología hace que sea propicio el uso del prototipo.

Principalmente el sistema tendrá que interactuar con la secretaria, se emitirán informes desde la dirección y por parte de cada uno de los profesores.

3.5. Identificar Casos de Uso y esquematizar Diagramas de Casos de Uso.

Un actor representa el rol genérico del sistema. El nombre que se le de a un actor deberá reflejar el papel que tendrá para el sistema.

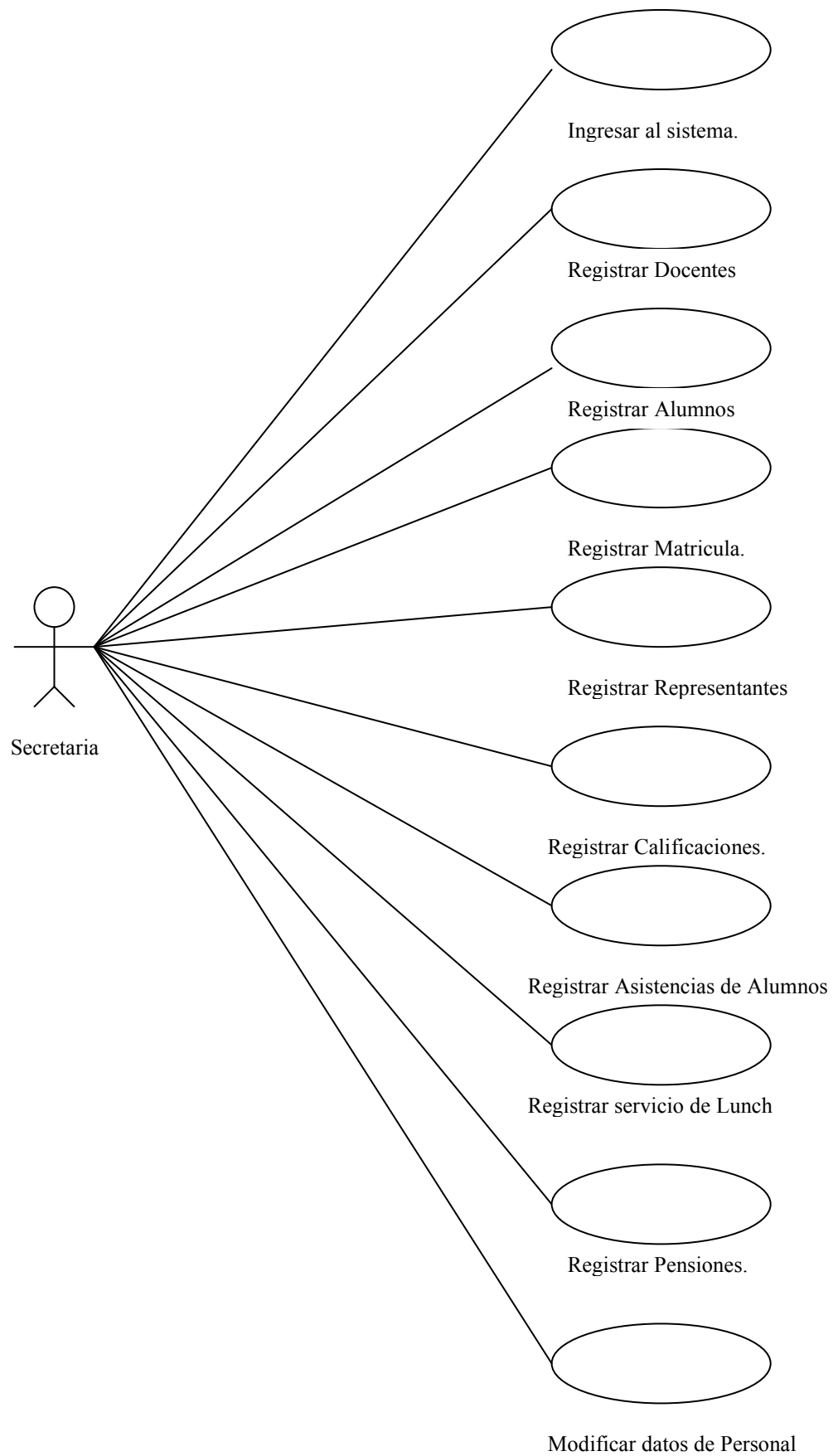
Los casos de uso son una descripción de un conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable de interés para un actor particular. Reflejan el uso que harán los actores del sistema; se muestran a través de ellos tanto las funcionalidades que ofrecerá el sistema, como los diferentes inherentes a las situaciones contempladas para cada una de estas.

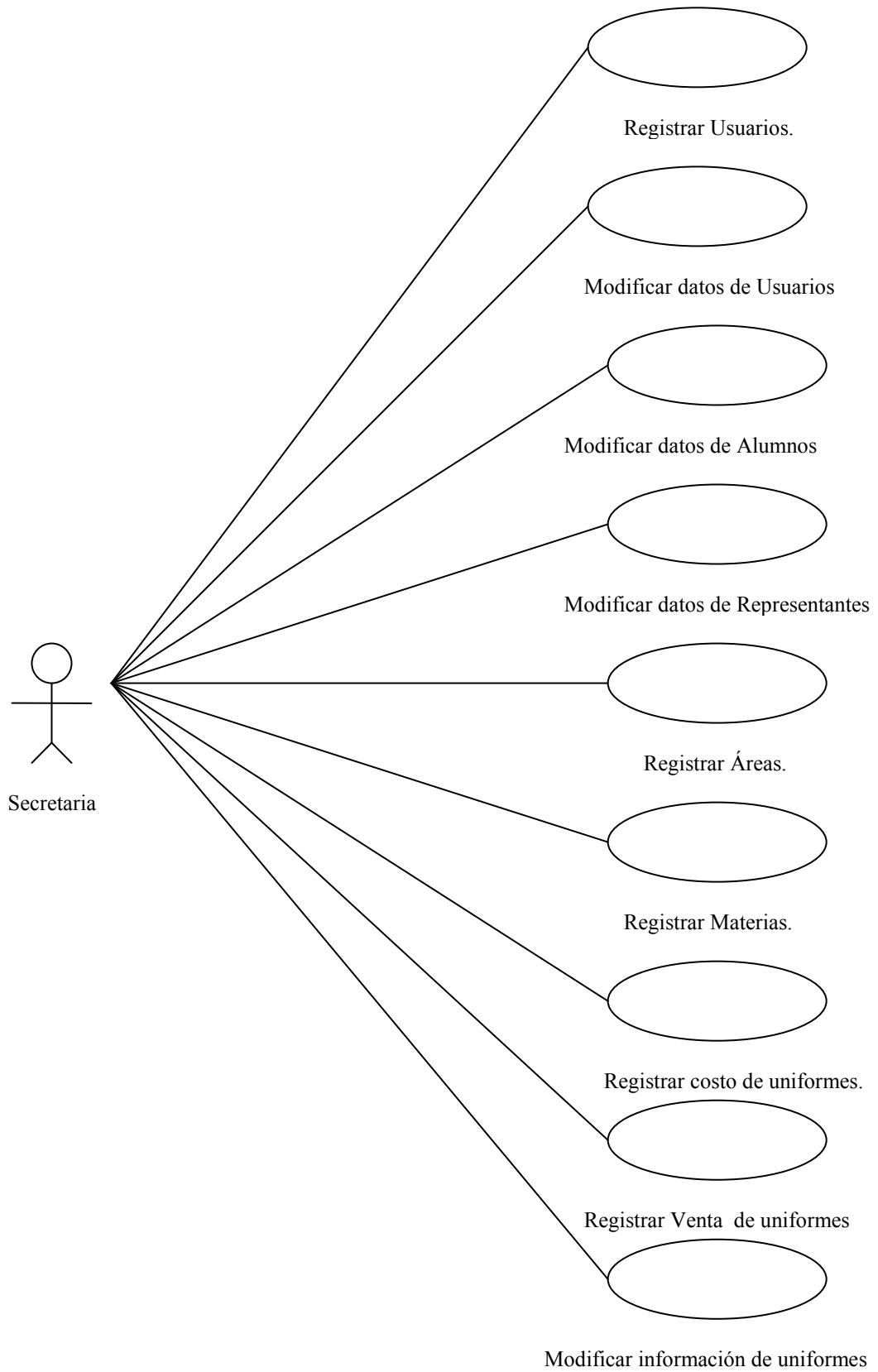
Actor	Caso de Uso
Secretaria	Ingresar al sistema. Registrar Personal. Registrar Alumnos. Registrar Matricula. Registrar Representante. Registrar Calificaciones. Registrar Asistencias de Alumnos.

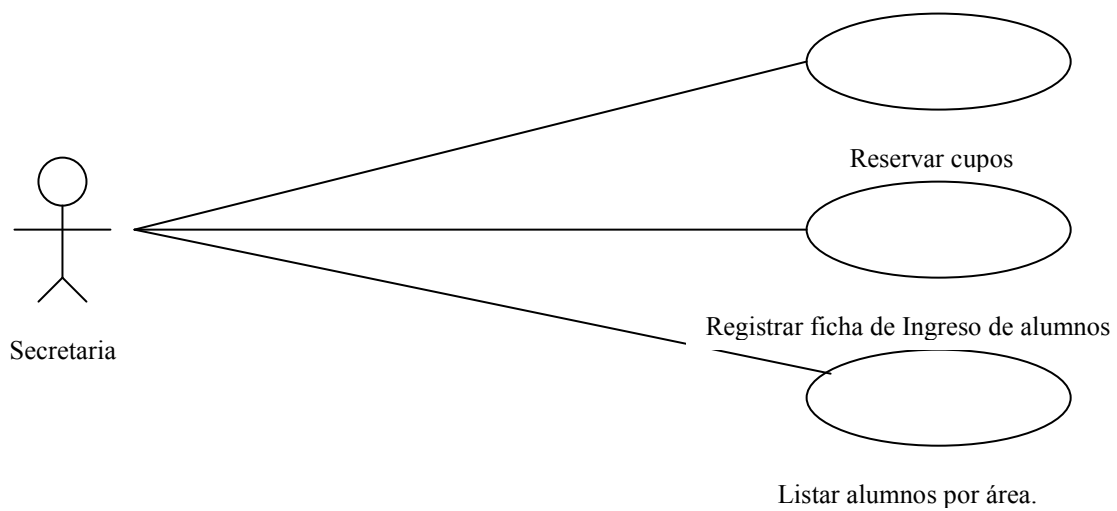
	Registrar servicio de Lunch. Registrar Pensiones. Modificar datos de los docentes. Modificar datos de Alumnos. Modificar datos de Representantes. Modificar datos de usuario_sistema. Registrar Áreas. Registrar Materias. Registrar costo de Uniformes. Modificar información de uniformes. Venta de Uniformes. Registrar pago por servicio de lunch.
--	---

Los diagramas de Casos de Uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción, con los usuarios y/o otros sistemas. O lo que es igual un diagrama que muestre la relación entre los actores y los casos de uso en un sistema.

Estos diagramas son los primeros en generarse ya que permiten capturar de forma sencilla las especificaciones del sistema a desarrollar. Además, ofrecen una visión del sistema, que vamos a desarrollar.







CAPITULO IV

ANÁLISIS Y DISEÑO

PRELIMINAR.

4.1 **Narración de los casos de Uso encontrados.**

Un caso de uso es una descripción de un conjunto de secuencias de acciones que el sistema ejecuta, que produce un resultado observable de interés para un actor particular, un caso de uso se utiliza para estructurar los aspectos de comportamiento en un modelo, los casos de sus se escriben con el fin de expresar lo que debe hacer el sistema a desarrollar, sin tener en cuenta como debe hacerlo.

A continuación se presentan los casos de uso del sistema Académico para el Centro de Desarrollo Infantil Bilingüe Angelitos de Luz.

Identificación	UC-001	
Caso de Uso.	Registrar Usuario.	
Actores:	Secretaria, usuario.	
Propósito:	Permitir a la secretaria dar acceso a un determinado grupo de personas el acceso al sistema, mediante un Login y un password.	
Descripción:	Este use case empieza cuando la secretaria se ubica en el menú usuarios, luego va al submenú nuevos Usuarios, el nuevo usuario registra su login y su password, este ultimo (password) escribiendo dos veces.	
Tipo:	Secundario.	
Referencia cruzada:	R.008	
Curso Típico de Eventos.		
Acción del Actor	Respuesta del Sistema.	

1. La secretaria se ubica en el menú Usuarios de la pantalla principal.	
2. La secretaria escoge el submenú nuevos usuarios.	3. El sistema presenta el formulario de registrar nuevos usuarios
4. El nuevo usuario ingresa un login y su contraseña, la contraseña la deberá repetir.	5. El sistema valida la información ingresada y crea una nueva cuenta de usuario del sistema.
	6. El sistema presenta un mensaje de que el usuario ha sido creado satisfactoriamente.
Curso alterno de acción.	
Paso 5: La información ingresada es incorrecta, el usuario ya esta creado, o su contraseña no son iguales, el sistema presentara un mensaje de error.	

Identificación	UC-002
Caso de Uso:	Registrar Áreas.
Actores:	Secretaria.
Propósito:	Permitir a la secretaria registrar las diferentes áreas de estudio con sus respectivos paralelos.
Descripción:	Este use case comienza cuando la secretaria ingresa los nombres de cada una de las áreas de estudio con sus respectivos paralelos.

Tipo:	Primario.	
Referencia cruzada.	R.020	
Curso típico de eventos.		
Acción del Actor		Respuesta del sistema.
1.	La secretaria se ubica en el menú Académico, escoge el submenú registrar áreas de estudio.	2. El sistema le presenta el formulario de registrar nuevas áreas de estudio.
3.	La secretaria ingresa la información necesaria y presiona el botón Guardar.	4. El sistema valida la información, crea una nueva área de estudio y presenta un mensaje que la nueva área se creó satisfactoriamente.
Curso alterno de Acción.		
Paso 4: Los datos ingresados están incompletos, o el área y el paralelo de estudio ya han sido creados con anterioridad		

Identificación	UC-003
Caso de Uso:	Registrar Materias.
Actores:	Secretaria.
Propósito:	Permitir a la secretaria registrar materias de estudio por área.

Descripción:	Este use case empieza cuando la secretaria se sitúa sobre el menú Académico, luego selecciona el submenú registrar materias, procede a llenar los campos y luego presiona el botón guardar.	
Tipo:	Primario.	
Referencia Cruzada:	R.021	
Curso típico de Eventos.		
Acción del Actor	Respuesta del Sistema.	
1. La secretaria se sitúa en el menú Académico, luego en el submenú registrar materias.	2. El sistema presentara un formulario, en el cual se pueda registrar las materias	
3. La secretaria ingresa las materias con sus respectivas áreas de estudio.	4. El sistema valida la información ingresada y crea una nueva materia.	
	5. El sistema presentara un mensaje que la materia se a agregado a su área satisfactoriamente.	
Curso alterno de Acción.		
Paso 5: La información ingresada es incorrecta o existen campos vacíos, el sistema presentara un mensaje de error.		

Identificación	UC-004
Caso de Uso:	Registrar Representante.
Actores:	Secretaria, representante.
Propósito:	Permitir guardar la información del padre del alumno.
Descripción:	Este use case empieza cuando la secretaria se sitúa en el menú Representante, luego en el submenú registrar datos, procede a llenar los campos de la información que e representante le haya dictado y presiona el botón guardar.
Tipo:	Primario.
Referencia Cruzada.	R.016
Curso típico de Eventos.	
Acción del Actor.	Respuesta del Sistema.
1. La secretaria se sitúa en el menú representante, luego en el submenú Registrar Datos.	2. El sistema presentara un formulario para registrar los datos del representante.
3. La secretaria pregunta la información al representante.	

<p>4 El representante le dicta a la secretaria sus datos personales.</p>	
<p>5. La secretaria procede a ingresar la información.</p>	<p>6. El sistema valida la información ingresada y crea un nuevo registro del representante.</p>
	<p>7. El sistema presentara un mensaje de que los datos del representante se ha almacenado satisfactoriamente.</p>
<p>Curso Típico de Eventos.</p>	
<p>Paso 7: La información ingresada es incorrecta, el sistema presentara un mensaje de advertencia.</p>	

Identificación	UC-005	
Casos de uso	Registrar ficha de Ingreso de alumnos.	
Actores	Secretaria, representante.	
Propósito	Conocer la información del niño.	
Descripción	Este use case empieza cuando la secretaria solicita información del niño al representante, para saber los datos generales del niño, como enfermedades, etc., además ver como se puede trabajar con el niño.	

Tipo	Primario.	
Referencia cruzada	R.025	
Curso típico de eventos		
Acción del actor	Respuesta del sistema.	
1. Este use case empieza cuando el representante acude a secretaria a matricular al niño.		
2. La secretaria se sitúa sobre el menú académico, luego en el submenú ficha de ingreso.	3. El sistema presentar el formulario ficha de ingreso.	
4. La secretaria solicita información del niño al representante.		
5. El representante informa a la secretaria los datos del niño, enfermedades, medicinas que usa, etc.		
6. La secretaria escribe la información.	7. El sistema presentara un mensaje de que la información del niño se registro satisfactoriamente.	
Curso alterno de acción		
Paso 6: La información ingresada es incorrecta, el sistema presenta un mensaje de advertencia.		

Identificación	UC-006
Caso de Uso:	Registrar Matricula.
Actores:	Secretaria, representante.
Propósito:	Capturar una matricula y su respectiva área.
Descripción:	La secretaria se sitúa sobre el menú Académico, luego en el submenú registrar matricula. La secretaria ingresa toda la información que el representante le proporciono y presiona el botón Guardar. El representante se marcha con un recibo de matricula.
Tipo:	Primario.
Referencia Cruzada:	R.022
Curso típico de Eventos.	
Acción del Actor	Respuesta del sistema
1. Este use case empieza cuando el representante llega secretaria y solicita una matricula para su representado.	
2. La secretaria le solicita los respectivos documentos para proceder a la matriculación.	

3. El representante entrega la información solicitada.	
4. La secretaria se sitúa sobre el menú Académico luego en el submenú registrar Matricula.	5. El sistema presentara un formulario de matricula
6. La secretaria ingresa la información del estudiante.	7. El sistema valida la información ingresada y crea una nueva matricula.
	8. El sistema presentara un mensaje de que la matricula se realizo satisfactoriamente.
9. La secretaria da al representante un recibo impreso.	
10. El representante se marcha con su comprobante de matricula.	

Identificación	UC-007
Casos de uso	Modificar datos de alumnos.
Actores	Secretaria, representante.
Propósito	Registrar información consistente del alumno.

Descripción	Este use case empieza cuando el representante le indica a la secretaria que existe nueva información del niño.	
Tipo.	Secundario.	
Referencia cruzada	R.014	
Curso típico de eventos		
Acción del actor	Respuesta del sistema.	
1. Este use case empieza cuando el representante acude a secretaria a entregar nueva información del niño.		
2. La secretaria se sitúa en el menú Estudiante, luego escoge el submenú modificar datos de alumno.	3. El sistema presentara el formulario modificar datos de alumno.	
4. La secretaria escribirá los apellidos y/o nombres del alumno.	5. El sistema presentara todos los nombres que coincidan.	
6. La secretaria seleccionara los apellidos y nombres del alumno, realizando doble clic.	7. El sistema habilitara la página con los datos del alumno a modificar.	
8. La secretaria procede a modificar los datos del alumno y presiona el botón	9. El sistema presentara un mensaje de que los cambios se realizaron satisfactoriamente.	
Curso alterno de acción		
Paso 4: La secretaria no ingreso los datos correctos.		

Identificación	UC-008
Caso de Uso:	Registrar Profesores.
Actores:	Secretaria, profesor.
Propósito:	Permitir almacenar los datos de los profesores.
Descripción:	Registrar la información de los profesores que laboran en la Institución, la secretaria registra la información del profesor.
Tipo:	Secundario.
Referencia cruzada	R.005
Curso típico de Eventos.	
Acción del Actor	Respuesta del sistema.
1. Este use case empieza cuando el profesor llega a secretaria para que registren sus datos.	
2. La secretaria se sitúa en el menú Profesor, luego en el submenú registrar profesor.	3. El sistema presentara el formulario para registrar los datos del profesor.
4. La secretaria solicita los datos al profesor.	
5. El profesor proporciona la información.	

<p>6. La secretaria ingresa la información en el formulario de registrar profesor.</p>	<p>7. El sistema valida la información ingresada y crea un nuevo registro del profesor.</p>
	<p>8. El sistema presentara un mensaje de que los datos se almacenaron satisfactoriamente.</p>
<p>Curso Alterno de Acción.</p>	
<p>Paso 8: la información ingresada es incorrecta, el sistema presentara un mensaje de error.</p>	

Identificación	UC-009
Casos de uso	Modificar datos de los docentes.
Actores	Secretaria, profesor.
Propósito	Modificar o agregar nueva información del profesor.
Descripción	Este use case empieza cuando el profesor acude a secretaria, para realizar una actualización o modificación de sus datos
Tipo	Secundario.
Referencia cruzada	R.015
<p>Curso típico de eventos</p>	

Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando el profesor va a secretaria para modificar sus datos.	
2. La secretaria se sitúa sobre el menú profesor luego se dirige al submenú modificar datos.	3. El sistema presentara un formulario Modificar datos del Docente.
4. La secretaria ingresa los apellidos del profesor.	5. El sistema presentar un listado de los nombres que coincidan con la información a modificar.
6. La secretaria da un doble clic sobre los datos a modificar.	7. El sistema habilitara la pagina de resultados.
8. La secretaria modifica o actualiza los datos del profesor y presiona el botón guardar..	9. El sistema presentara un mensaje de que los cambios se realizaron satisfactoriamente.
Curso alterno de acción	
Paso 4: Los apellidos no coinciden con los datos almacenados.	

Identificación	010
Casos de uso	Modificar datos de Representantes.
Actores	Secretaria, representante.
Propósito	Modificar los datos del representante.
Descripción	Este use case empieza cuando el representante de un alumno acude a secretaria para informar a la secretaria de sus nuevos datos.
Tipo	Secundario.

Referencia cruzada	R.016
Curso típico de eventos	
Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando el representante va a secretaria a informar que existen nueva información con respecto a el.	
2. La secretaria realiza clic sobre el menú representante, luego sobre el submenú modificar datos.	3. El sistema presentara un formulario modificar datos del representante.
4. La secretaria ingresara los apellidos del representante.	5. El sistema presentara un listado de apellidos que coincidan con los del representante.
6. La secretaria realizara doble clic sobre los apellidos a modificar.	7. El sistema habilitara la pagina de modificar.
8. La secretaria procede a modificar los datos y presiona el botón guardar.	9. El sistema presentara un mensaje de que los cambios efectuados se realizaron satisfactoriamente.
Curso alterno de acción	
Paso 4: Los apellidos no coinciden con los datos almacenados.	

Identificación	UC-011
Casos de uso	Modificar datos usuarios_ sistema.
Actores	Usuario_ sistema.
Propósito	Modificar las cuentas de los usuarios que tienen permiso

	de manejar el sistema.	
Descripción	Este use case empieza cuando el usuario_sistema, acude a secretaria para modificar su cuenta.	
Tipo	Primario.	
Referencia cruzada	R.026	
Curso típico de eventos		
	Acción del actor	Respuesta del sistema.
	1. Este use case empieza cuando el usuario_sistema se sitúa en el menú usuarios, luego en el submenú modificar datos.	2. El sistema presentara el formulario modificar datos usuario_sistema.
	3. El usuario_sistema ingresara sus datos, y presionara el botón buscar.	4. El sistema habilitara la pagina resultados.
	5. El usuario_sistema ingresara sus nuevos datos, y presionara el botón guardar..	6. El sistema presentara un mensaje que los cambios se efectuaron satisfactoriamente.
Curso alternativo de acción		
Paso 3: El usuario_sistema ingresa datos inválidos, el sistema presentar un mensaje de advertencia.		

Identificación	UC-012
Caso de Uso:	Registrar pago de pensiones.
Actores:	Secretaria, representante.
Propósito:	Capturar el pago de pensiones.

Descripción:	Este use case empieza cuando el representante llega a cancelar la mensualidad de su representado, en secretaria.	
Tipo:	Primario.	
Referencia cruzada.	R.023	
Curso típico de Eventos.		
Acción del actor	Respuesta del sistema.	
1. Este use case empieza cuando el representante llega a secretaria a cancelar la pensión de su representado.		
2. La secretaria hace clic sobre el menú Académico luego se dirige al submenú, pago de pensiones.	3. El sistema presentara el formulario, pago de pensiones.	
4. La secretaria procede a ingresar la fecha de pago.		
5. La secretaria solicita el Número de Identificación del representante, y los nombres y apellidos del representado.		

6. El representante procede a dictarle su número de identificación, los nombres de su representado.	
7. La secretaria ingresa la información y presiona el botón Guardar.	8. El sistema buscará si los datos están del representante y representado están registrados.
9. La secretaria presiona el botón Imprimir	10. El sistema imprimirá un recibo.
Curso alternativo de acción.	
Paso 8: Los datos del representante y del representado no constan en la base de datos.	

Identificación	UC-013
Caso de Uso:	Reservar cupo.
Actores:	Secretaria, representante.
Propósito:	Reservar cupo en una determinada área de estudio para un alumno.
Descripción:	Este use case empieza cuando el representante llega a secretaria para reservar un cupo para su representado.
Tipo:	Secundario.

Referencia Cruzada:	R.024
Curso típico de eventos.	
Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando el representante llega a secretaria, para reservar un cupo para su representado.	
2. La secretaria hace clic sobre el menú Académico, luego se desplaza al submenú reservar cupos.	3. El sistema presentara un formulario para reservar cupos.
4. La secretaria solicita al representante información del niño, y área de estudio.	
5. El representante dicta los datos de su representado.	
6. La secretaria ingresa los datos del infante y al área de estudio a la cual ingresara.	
7. La secretaria solicita al representante su número de identificación.	

8. El representante le dicta a la secretaria su número de identificación.	
9. La secretaria escribe el número de identificación del representante, y presiona el botón Guardar.	10. El sistema presentara un mensaje de que la reservación se efectuó satisfactoriamente.
Curso alterno de acción.	
Paso 9: El sistema presentara un mensaje de error de que la información ingresada no es correcta o existen campos vacíos.	

Identificación	UC-014
Caso de Uso:	Pago servicio de Lunch.
Actores:	Secretaria, representante.
Propósito:	Registrar al infante para que reciba alimentos a una determinada hora del día.
Descripción:	Este use case empieza cuando el representante llega a secretaria, para que su hijo sea inscrito en el servicio de lunch.
Tipo:	Secundario.
Referencia cruzada.	R.006
Curso típico de eventos.	

Acción del actor.	Respuesta del sistema.
1. Este use case empieza cuando el representante llega a secretaria para inscribir a su hijo en el servicio de lunch.	
2. La secretaria hace clic sobre el menú varios luego se dirige al submenú cobrar servicio de Lunch.	3. El sistema presentara un formulario de pago por el servicio de Lunch
4. La secretaria procede a ingresar la fecha de pago.	
5. La secretaria solicita el Número de Identificación del representante, y los nombres y apellidos del representado.	
6. El representante procede a dictarle su número de identificación, los nombres de su representado.	
7. La secretaria procederá a ingresar el costo y el i.v.a.	8. El sistema calculara el total a pagar.

9. La secretaria presionara el botón Guardar.	10. El sistema buscara si los datos del representante y representado están registrados.
11. La secretaria presiona el botón Imprimir.	12. El sistema imprimirá un recibo.
Curso alternativo de acción.	
Paso 8: El sistema presentara un mensaje de error, el campo costo es menor o igual a cero.	
Paso 10: El sistema presentara un mensaje de error, los datos no están almacenados en la Base de Datos.	

Identificación	UC-015
Caso de Uso:	Registrar Asistencias de Alumnos
Actores:	Secretaria, docente.
Propósito:	Llevar un registro de los días que el estudiante a asistido a clases.
Descripción:	Este use case empieza cuando el docente llega a secretaria a entregar el listado de asistencias que lleva, y la secretaria procede a automatizarlas.
Tipo:	Primario.
Referencia Cruzada:	R.019
Curso típico de Eventos.	

Acción del actor.	Respuesta del sistema.
1. Este use case empieza cuando el docente llega a secretaria para entregar el listado de asistencias de los alumnos.	
2. La secretaria realiza clic sobre el menú Académico, luego se desplaza al submenú Asistencias.	3. El sistema presentara un formulario de registro de asistencias de alumnos.
4. La secretaria escogerá el área de estudio, y hará doble clic sobre los nombres del niño.	5. El sistema habilitara la página de registros de asistencias.
6. La secretaria escribirá la fecha que el alumno no asistió, y presionara el botón guardar.	7. El sistema presentara un mensaje de que la fecha se almaceno satisfactoriamente.
<p style="text-align: center;">Curso alterno de acción.</p> <p>Pasó 6: La secretaria ingresa una fecha valida, el sistema presentara un mensaje de error fecha inválida.</p>	

Identificación	UC-016
Casos de uso	Registrar calificaciones.
Actores	Secretaria, profesor.
Propósito	Ingresar al sistema las notas que tiene cada alumno
Descripción	Este use case empieza cuando el profesor acude a secretaria para entregar las calificaciones de los alumnos.
Tipo	Primario.
Referencia cruzada	R.004
Curso típico de eventos	
Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando el profesor llega a secretaria para entregar las calificaciones de los alumnos.	
2. La secretaria da un clic sobre el menú académico, luego escoge el submenú registrar notas.	
	3. El sistema presentara un formulario para registrar notas.
4. La secretaria escogerá el área y paralelo, luego hará doble clic sobre los nombres del alumno	5. El sistema habilitara la pagina de registrar notas de alumnos.
6. La secretaria procede a ingresar las calificaciones del alumno, y presionara el botón guardar.	7. El sistema presentara un mensaje de que el registro de calificaciones se realizo

	satisfactiamente.
Curso alterno de acción	
Paso 6: La secretaria ingreso mal las calificaciones, el sistema presentara un mensaje de error.	

Identificación	UC-017
Casos de uso	Listar alumnos por área.
Actores	Secretaria.
Propósito	Presentar los alumnos matriculados en una determinada área de estudio.
Descripción	Este use case empieza cuando la secretaria se sitúa sobre el menú ver, luego sobre el submenú alumnos por área.
Tipo	Primario.
Referencia cruzada	R.009

Curso típico de eventos	
Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando la secretaria se sitúa sobre el menú ver, luego va al submenú alumnos por área.	2. El sistema presentara un formulario, para ver la nomina de alumnos por área.
3. La secretaria escogerá el área y paralelo de estudio, luego presionara el botón buscar.	4. El sistema presentara un listado de alumnos matriculados en dicha área, y presentara los nombres del profesor asignado.
5. La secretaria presionara el botón	6. El sistema generara un reporte

imprimir.	con el listado de alumnos, y nombres del profesor.
Curso alternativo de acción	
Paso 3: La secretaria no escogió el área de estudio, el sistema presentara un mensaje de advertencia.	

Identificación	UC-018
Casos de uso	Registrar uniformes.
Actores	Secretaria.
Propósito	Ingresar detalle de los uniformes.
Descripción	Este use case empieza cuando la secretaria registra la cantidad, talla, etc., de uniformes.
Tipo	Primario.
Referencia cruzada	R.027
Curso típico de eventos	
Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando la secretaria se sitúa sobre el menú varios, luego sobre el submenú registrar uniformes.	2. El sistema presentara un formulario Registrar Uniformes.
3. La secretaria procede a llenar los campos, para registrar uniformes, y presiona el botón calcular.	4. El sistema calcula el valor que tiene cada uniforme.
5. La secretaria presiona el botón	6. El sistema presentara un

guardar	mensaje de que los datos se almacenaron satisfactoriamente.
Curso alternativo de acción	
Paso 3: Los datos ingresados no son validos, el sistema presentara un mensaje de advertencia.	

Identificación	UC-019
Casos de uso	Registrar venta de uniformes.
Actores	Secretaria, representante.
Propósito	Registrar la venta de uniformes diario, cultura física.
Descripción	Este use case empieza cuando el representante y profesor acuden a secretaria para adquirir los uniformes.
Tipo	Primario.
Referencia cruzada	R.028
Curso típico de eventos	
Acción del actor	Respuesta del sistema.
1. Este use case empieza cuando el representante acude a secretaria para adquirir los uniformes.	
2. La secretaria se sitúa sobre el menú Varios, luego hace clic sobre el submenú venta de uniformes.	3. El sistema presentara un formulario venta de uniformes.

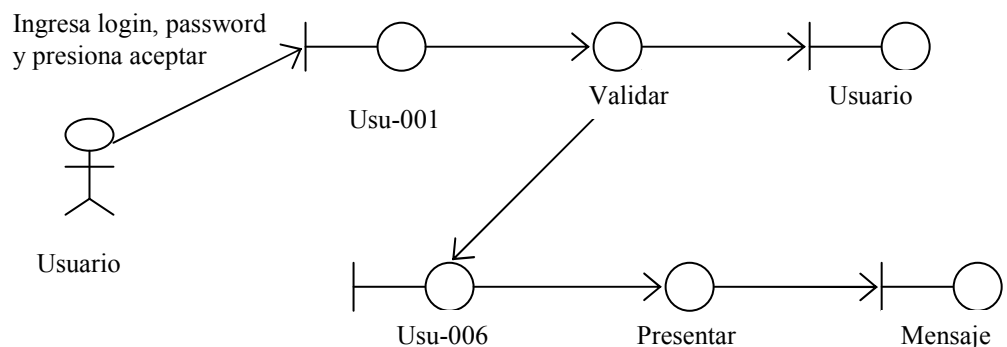
4. La secretaria ingresa la identificación del representante, datos de alumno, datos del uniforme; luego presiona el botón calcular.	5. El sistema calculara el total a pagar y presentara el resultado.
6. La secretaria presionara el botón guardar.	7. El sistema restara la cantidad vendida y presentara un mensaje de que se la transacción se realizo satisfactoriamente.
8. La secretaria presionara el botón imprimir.	9. El sistema generara un reporte con la información respectiva.
Curso alternativo de acción	
Paso 4: Los datos ingresados no son validos, el sistema presentara un mensaje de advertencia.	

Identificación	UC-020
Casos de uso	Modificar información de uniformes.
Actores	Secretaria.
Propósito	Modificar datos de los uniformes.
Descripción	Este use case empieza cuando la secretaria se sitúa sobre el menú varios, luego sobre el submenú modificar datos de uniformes.
Tipo	Secundario.
Referencia cruzada	R.029
Curso típico de eventos	
Acción del actor	Respuesta del sistema.

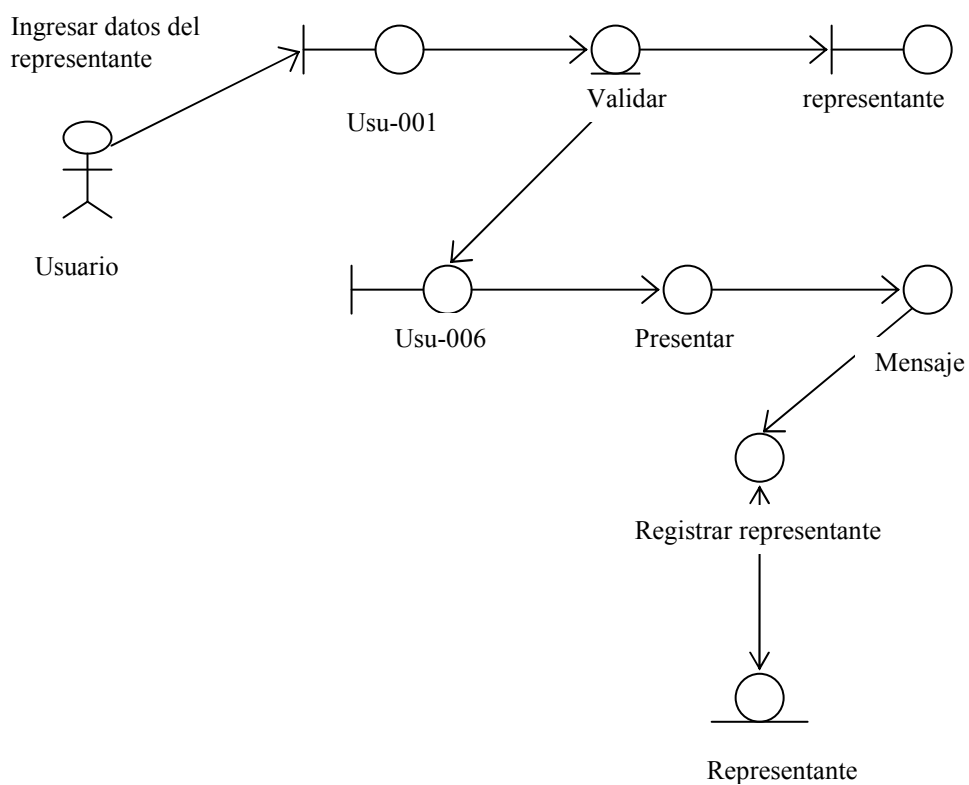
<p>1. Este use case empieza cuando la secretaria se sitúa sobre el menú varios, luego da clic sobre el submenú modificar datos de uniformes.</p>	<p>2. El sistema presentara un formulario Modificar datos de uniformes.</p>
<p>3. La secretaria selecciona y realiza doble clic sobre la información a modificar.</p>	<p>4. El sistema habilitara la pagina para modificar la información.</p>
<p>5. La secretaria ingresa la nueva información y presionara el botón guardar.</p>	<p>6. El sistema presentara un mensaje que los datos se almacenaron satisfactoriamente.</p>
<p>Curso alternativo de acción</p>	
<p>Paso 5: Los datos ingresados no son validos, el sistema presentara un mensaje de advertencia.</p>	

4.2 **Análisis de Robustez del Sistema.**

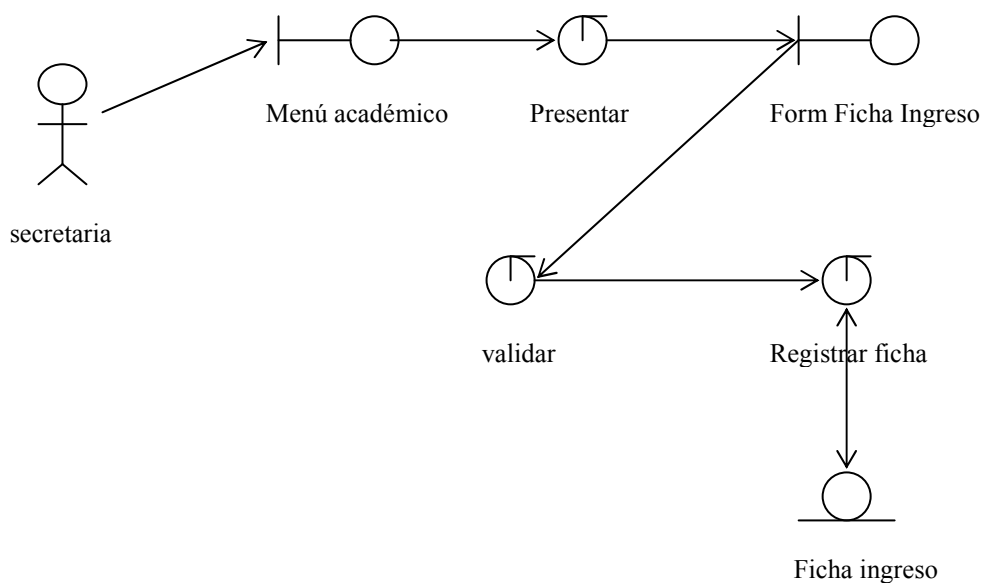
Ingresar al sistema.



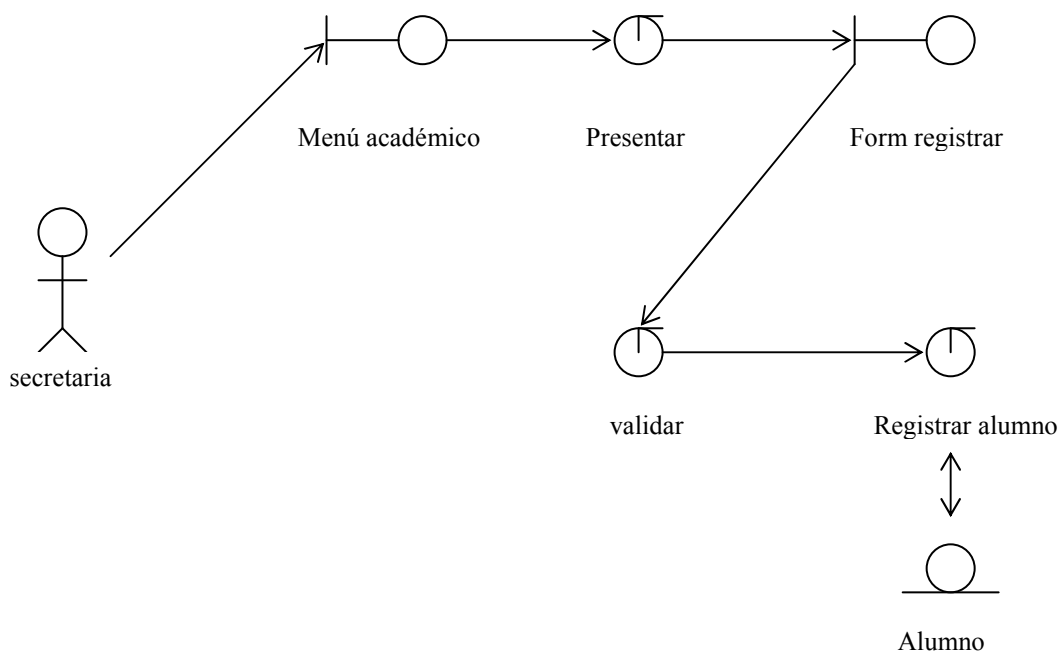
Registrar Representante.



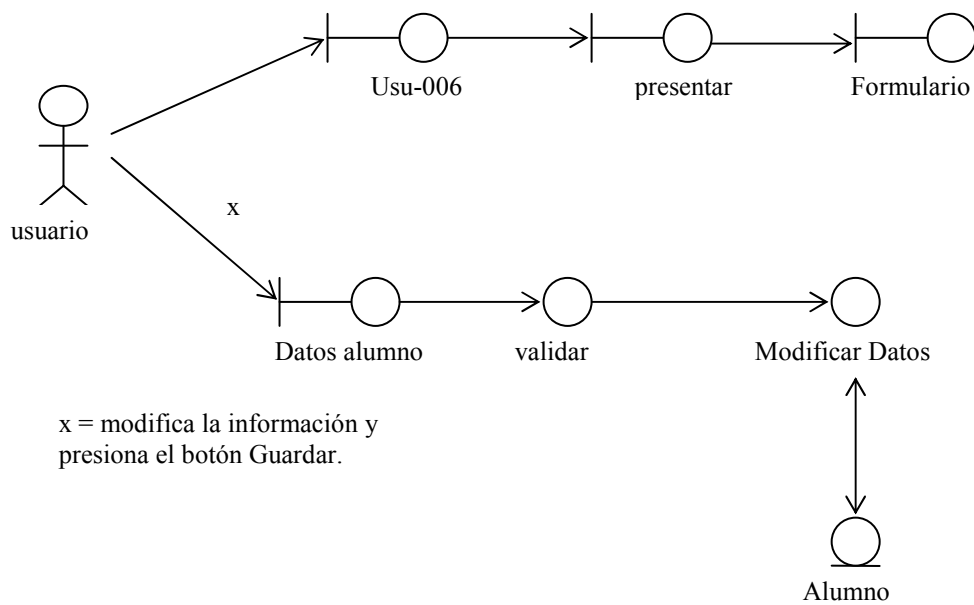
Registrar Ficha Ingreso Alumno.



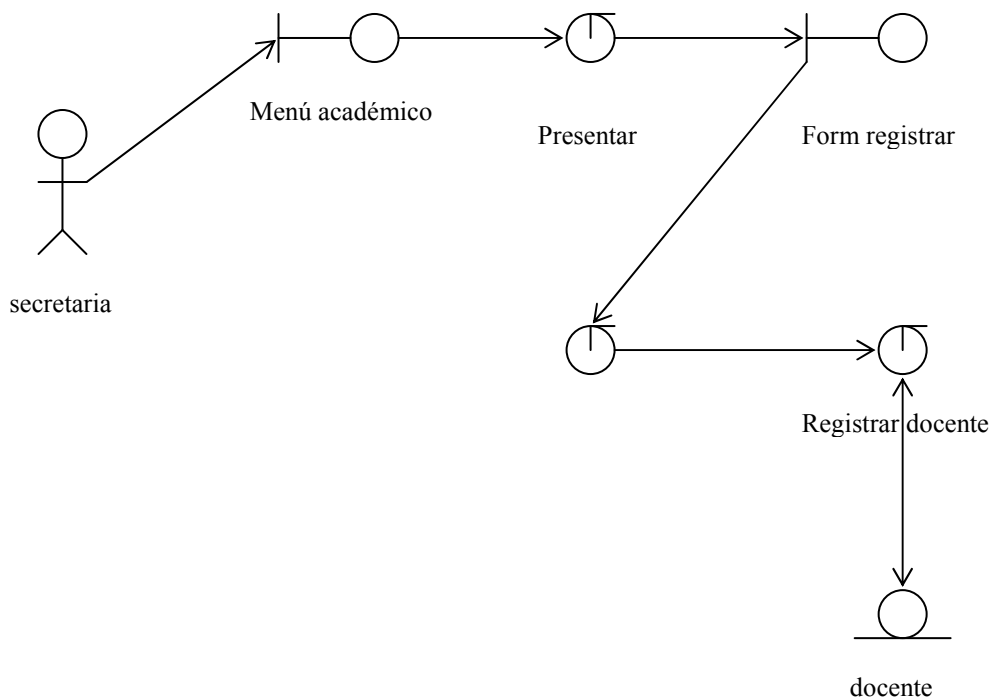
Registrar Alumno.



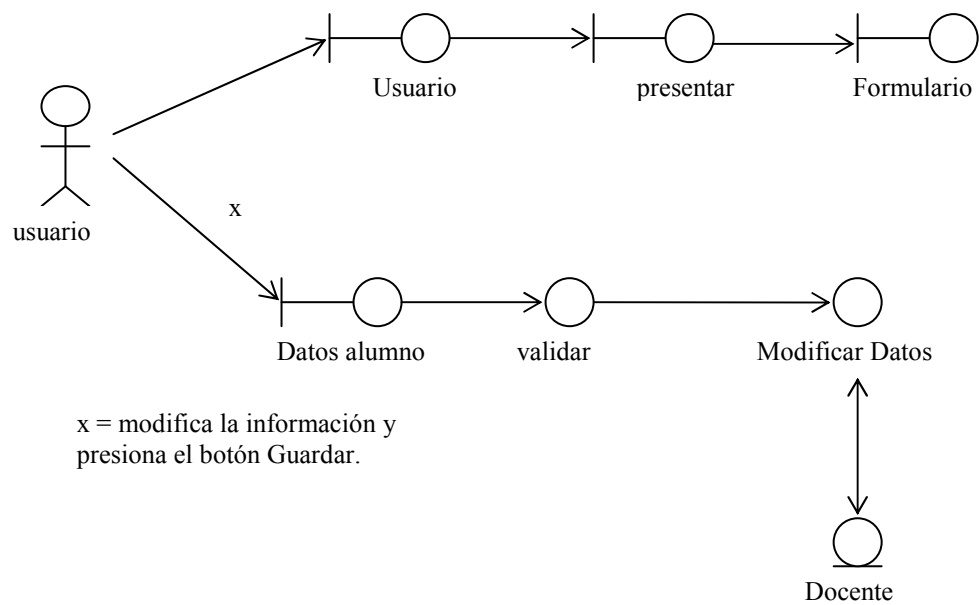
Modificar Datos Alumno.



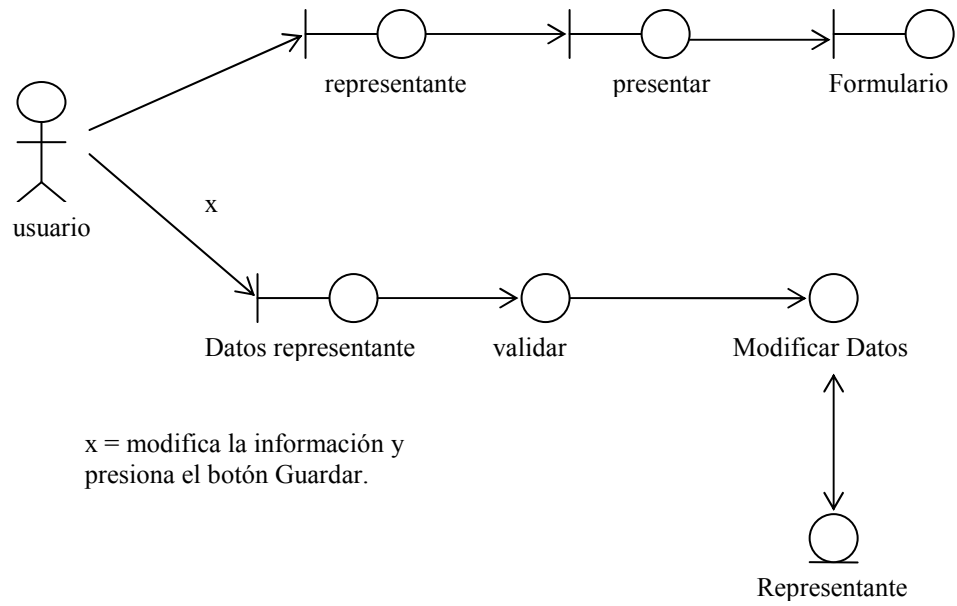
Registrar Docente.



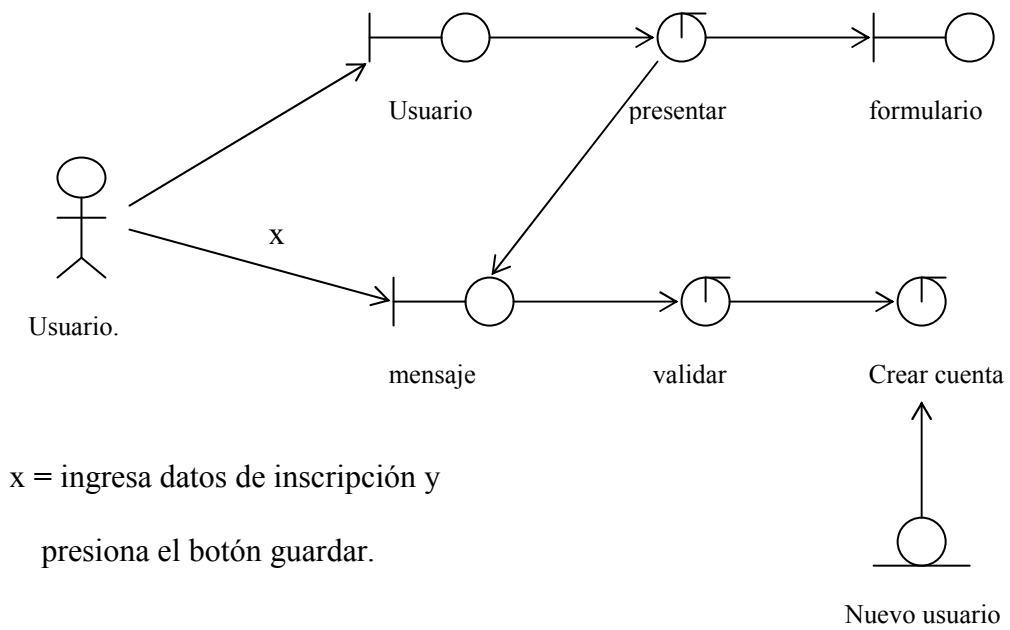
Modificar datos Docente.



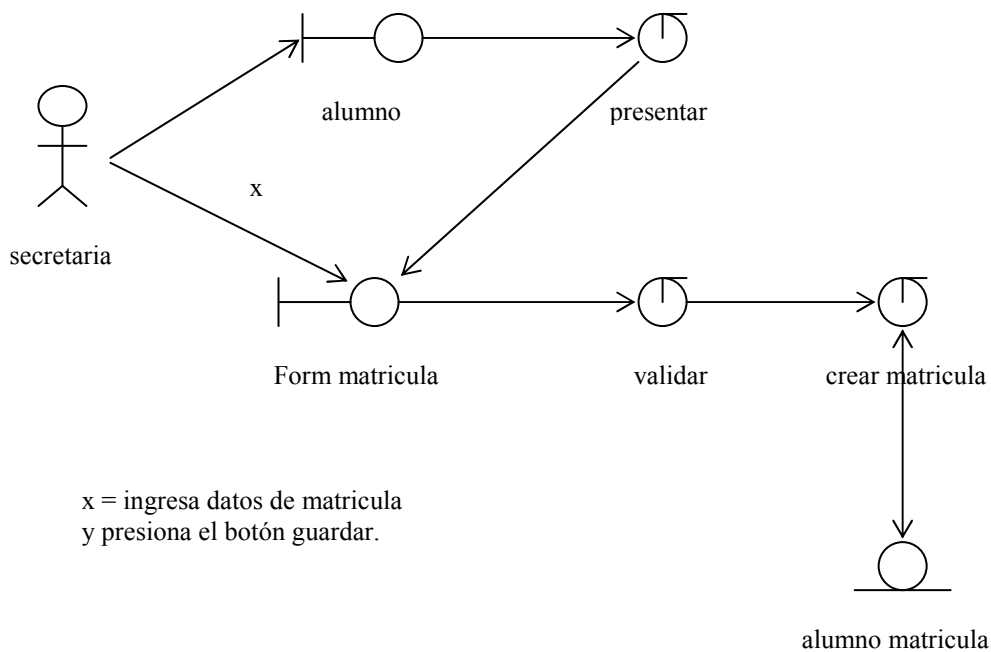
Modificar datos Representante.



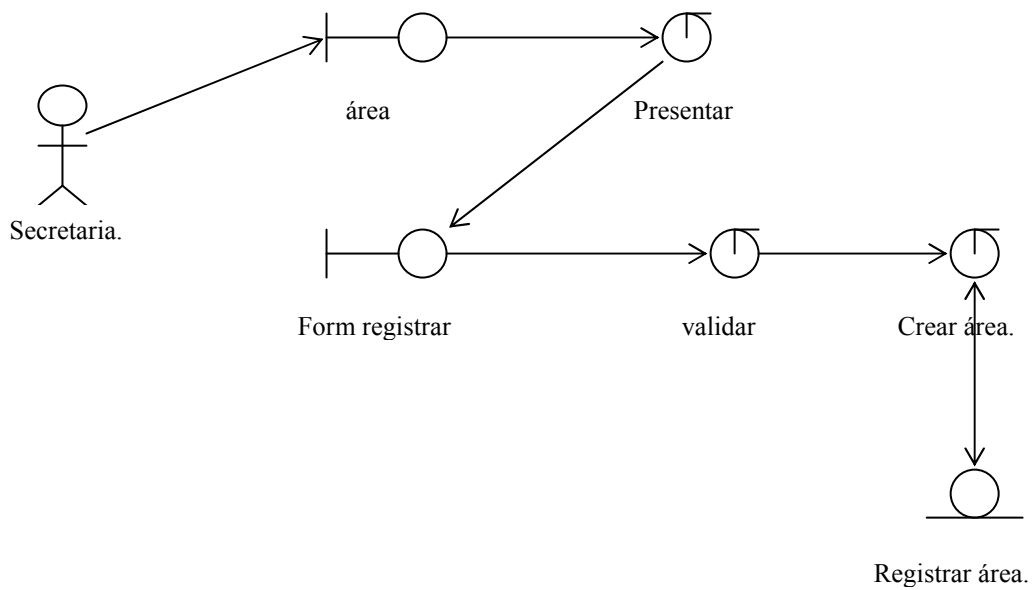
Registrar nuevo usuario_sistema.



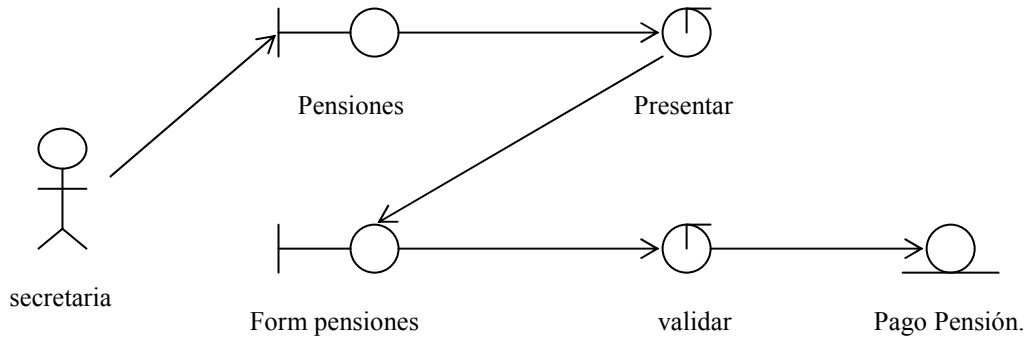
Registrar Matricula.



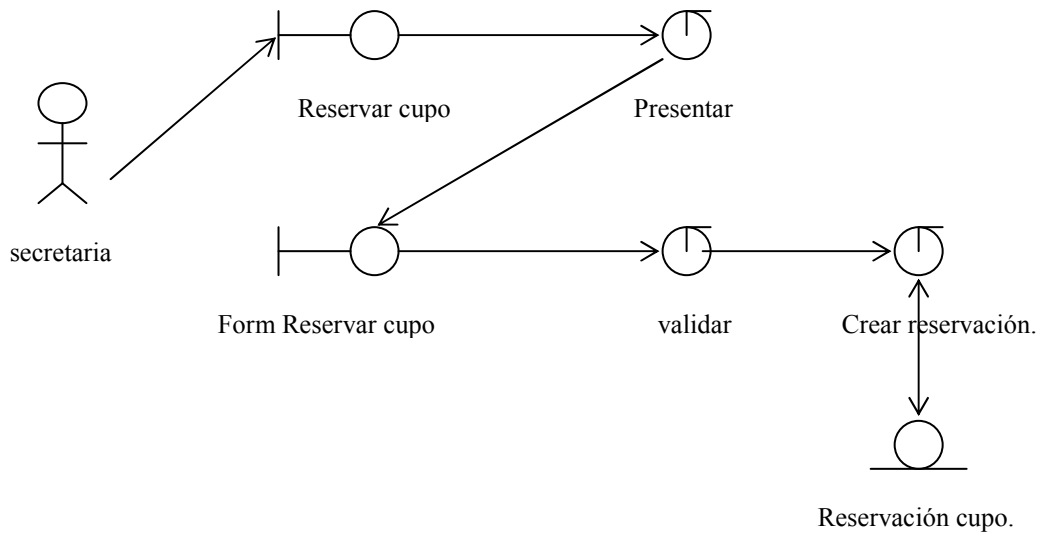
Registrar Áreas.



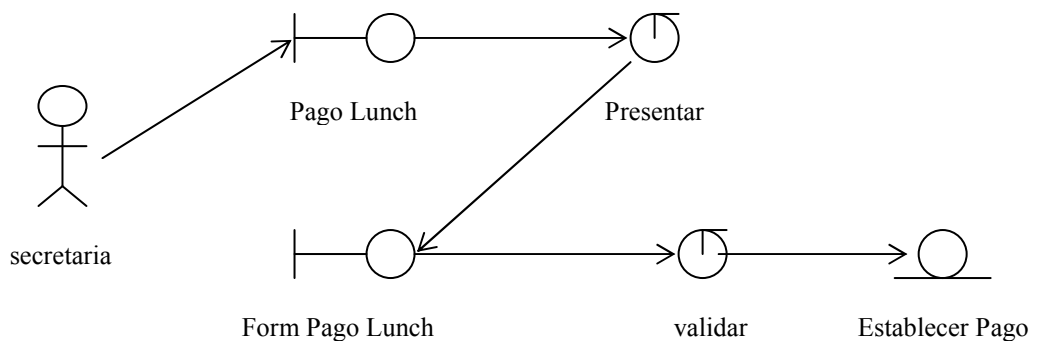
Registrar Pensiones.



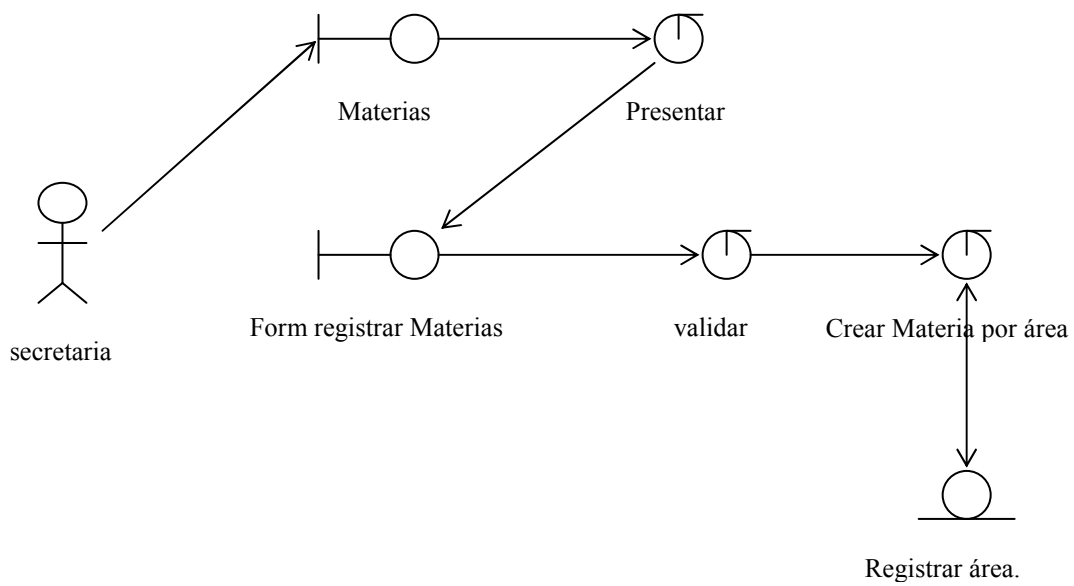
Reservar cupo.



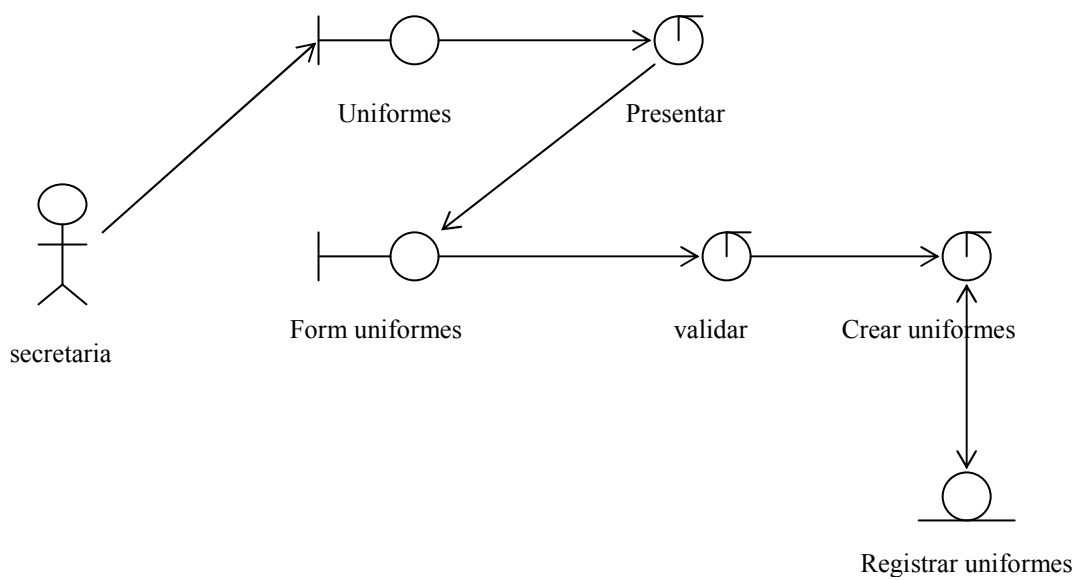
Pago Lunch.



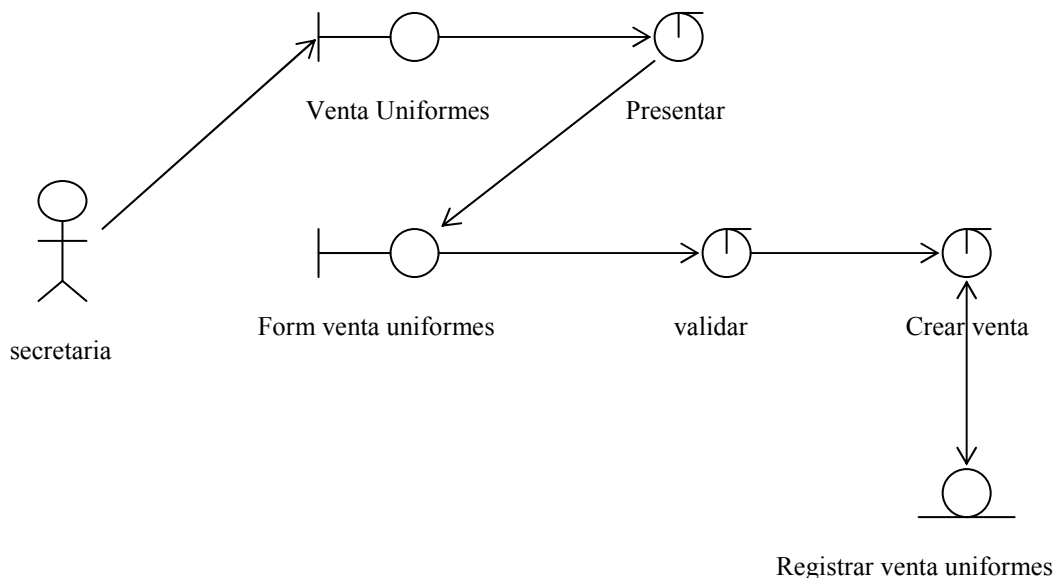
Registrar Materias.



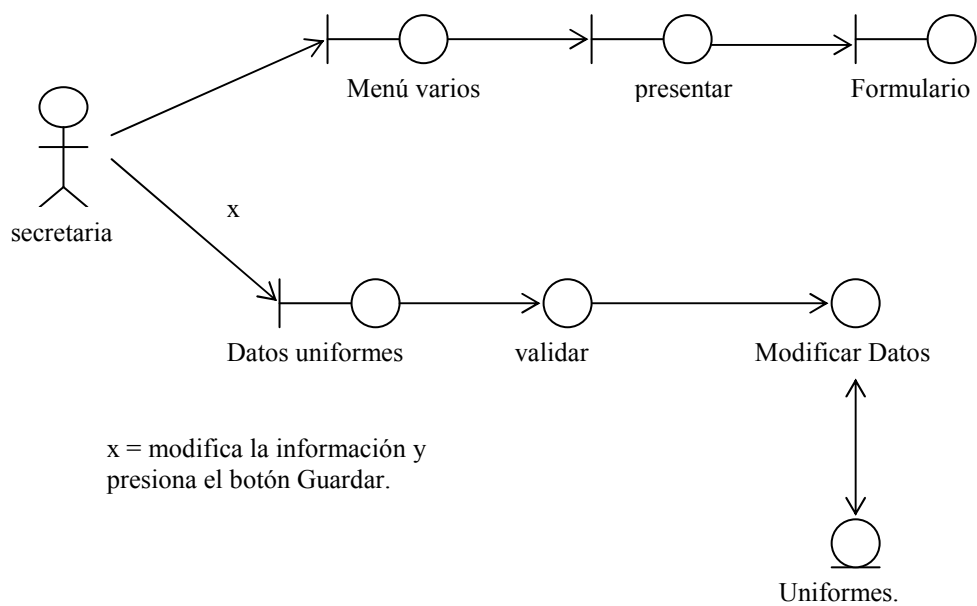
Registrar uniformes.



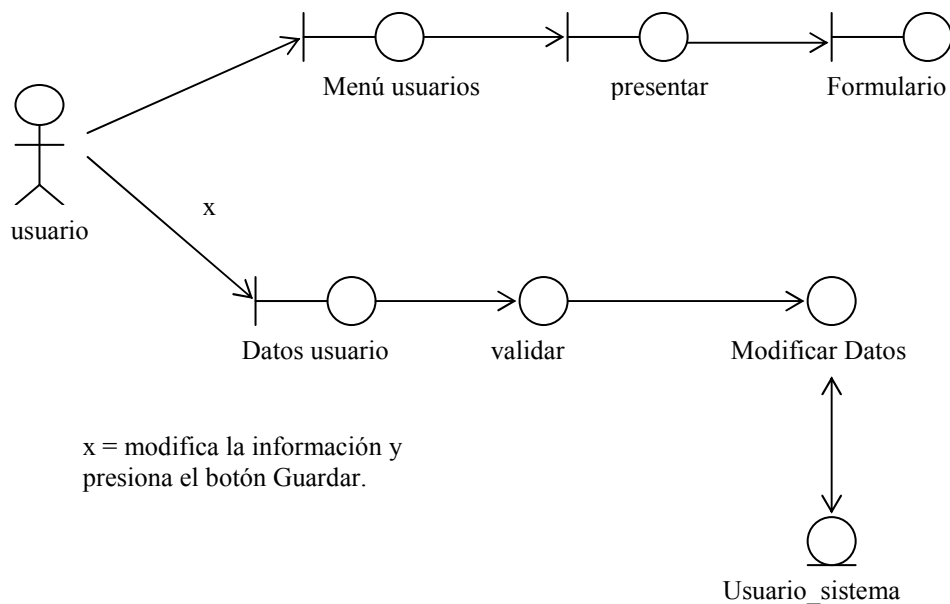
Venta Uniformes.



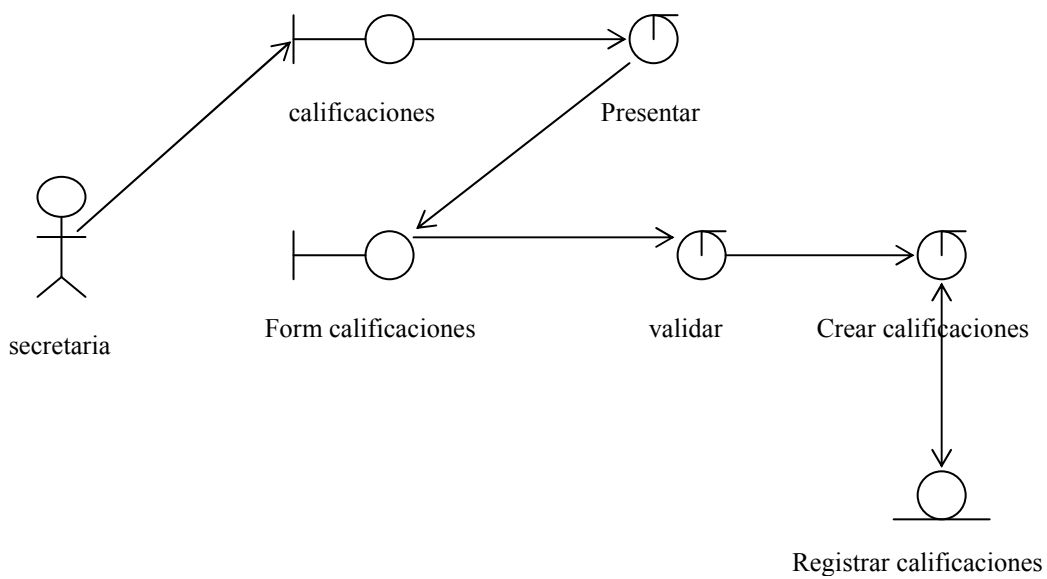
Modificar uniformes.



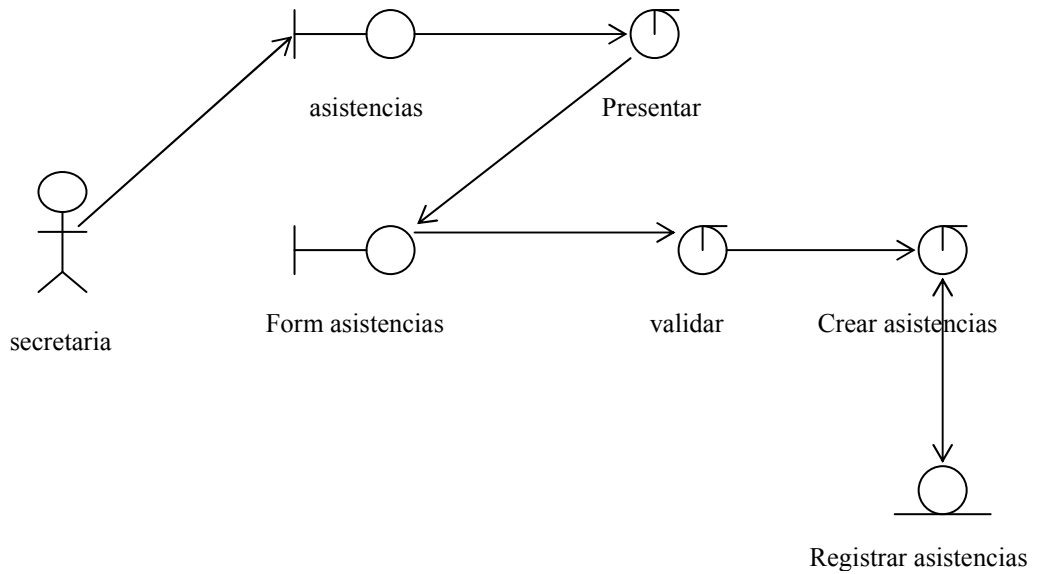
Modificar datos usuario.



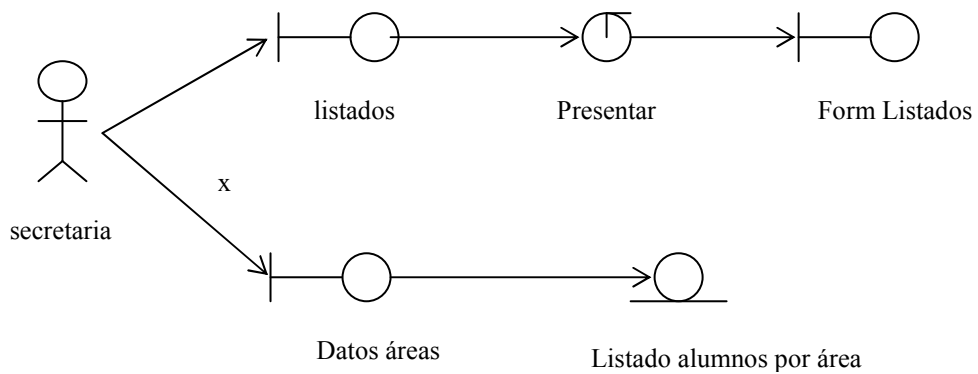
Registrar calificaciones.



Registrar asistencias Alumnos.



Listar alumnos por área.



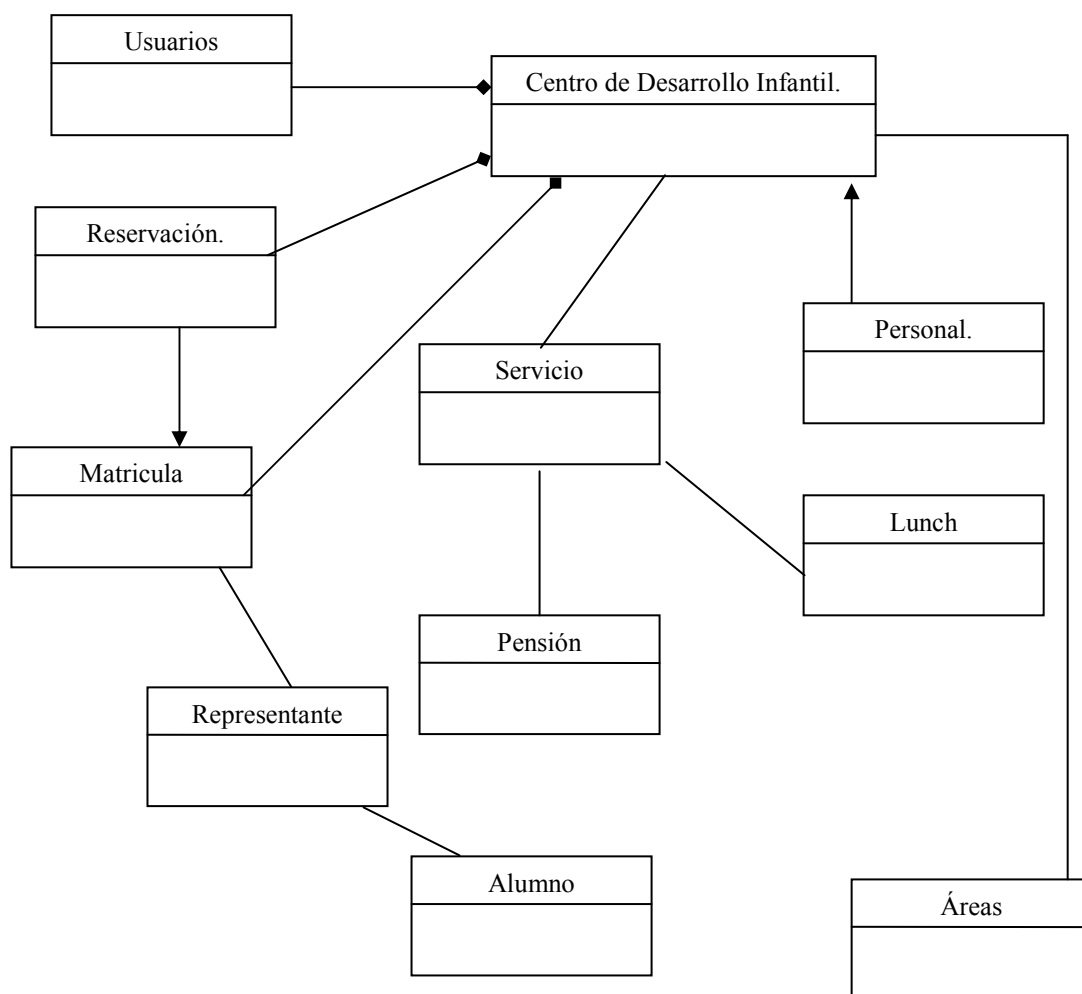
x = busca la información y presiona el botón imprimir.

4.2.1 Identificación de Objetos que intervienen en los escenarios.

- UsuarioSistema.
- Docente.
- Alumno.
- Matricula.

- Áreas.
- Materias.
- Pensiones.
- Representantes.
- Matricula.
- Pensión.
- Secretaria.

4.2.2 Actualización del modelo del dominio.



4.3 **Construcción de la Base de Datos, describiendo cada uno de sus campos.**

Tabla Usuarios.

Esta tabla almacenara toda la información acerca de los usuarios que tiene permiso para acceder al sistema.

Campo	Tipo Dato	Descripción
Id	Entero 4	Código de usuario
Nombre	Carácter 20	Nombre de usuario, que tiene acceso al sistema
Contraseña	Carácter 20	Permiso para acceder al sistema.

Índices tipo

Id primario

Tabla códigos.

Esta tabla almacenara el nombre del resto de las tablas del sistema para generar automáticamente el código principal de las mismas.

Campo	Tipo Dato	Descripción
Id	Entero 4	Código de incremento.
NombTable	Carácter 50	Nombre de la tabla.

Índices tipo

Id Normal.

Tabla áreas.

Esta tabla almacena los datos de las áreas y paralelos que existen en el centro.

Id	Integer 4	Código de incremento
Nombreaea	Carácter 24	Nombre área de estudio
Paralelo	Carácter 2	Asignación de área.

Índices.

Tipo

Id Principal

Tabla profesor

Almacena la información de los docentes que laboran en la institución.

Id	Integer 4	Código de incremento
Apellido	Character 40	Apellidos de los docentes
Nombre	Character 40	Nombres de los docentes.
fecNac	Date 8	Fecha de nacimiento del profesor
circupas	Character 13	Numero único de identificación del profesor
tipsangre	Character 5	Tipo de sangre del profesor.

Índices

Tipo

Id primary

Apellido regular.

Tabla detalleprof

Almacena la información personal del profesor

Id	Integer 4	Código de incremento
Codprof	Integer 4	Código del profesor
Telef1	Character 9	Teléfono de casa del docente
Telef2	Character 9	Teléfono del lugar donde labora el docente
Movil	Character 10	Celular del profesor.
Dirección	Character 50	Lugar donde se encuentra la casa del profesor.
Naturalde	Character 60	Lugar de nacimiento del docente.

Índices

Tipo

Id	primary
Codprof	regular

Tabla alumno

Almacena la información acerca del infante.

Id	Integer 4	Código de incremento.
Idrepresen	Integer 4	Código del representante
Apellidos	Character 30	Apellidos del alumno
Nombres	Character 30	Nombres del alumno
Sexo	Logical 1	Sexo del alumno

fecNac	Date 8	Fecha de nacimiento del alumno
tiposangre	Character 4	Tipo de sangre del alumno
edad	Integer 4	Edad en años del alumno

Índices.	Tipo
Id	primary
Idrepresen	regular
Apellidos	regular

Tabla detallealumno

Almacena la información personal del niño.

Id	Integer 4	Código de incremento
Idalumno	Integer 4	Código del alumno
Miedos	Memo 4	Temores del niño
Dirección	Character 52	Lugar donde se encuentra la vivienda del niño
Nacidoen	Character 35	Lugar de nacimiento del niño
Medicinas	Memo 4	Tipo de medicación que utiliza el niño
Alergias	Memo 4	A que tiene alergias el infante
Nombresponde	Character 20	Nombre al cual el niño entiende
Enfermedades	Memo 4	Enfermedades que el niño a

		padecido.
--	--	-----------

Índices	Tipo
Id	Primary
Idalumno	regular

Tabla representante

Esta tabla contiene información acerca del representante

Id	Integer 4	Código de incremento.
Cirucpas	Character 13	Numero de identificación del representante
Apellidos	Character 25	Apellidos del representante.
Nombres	Character 25	Nombres del representante.
Edad	Integer 4	Edad del representante.
Sexo	Logic	Masculino, femenino.
Tiposangre	Character 4	Tipo de sangre del representante.

Índices	tipo
Id	primary
Cirucpas	regular
Apellidos	regular

Tabla detallerepresentante.

Esta tabla contiene información personal acerca del representante.

Id	Integer 4	Código de incremento
Idrepres	Integer 4	Código del representante
Dirección	Character 50	Lugar donde se encuentra la casa del representante.
Ocupación	Character 43	Que tipo de empleo tiene el representante.
Telef1	Character 9	Numero de teléfono de la casa del representante.
Telef2	Character 9	Numero de teléfono del lugar donde labora o se encuentra el representante.
Móvil	Character 10	Celular del representante.

Índices	tipo
Id	primary
Idrepres	regular.

Tabla Cupos

Esta tabla contiene información de los alumnos que se inscriben en una determinada área de estudio.

Id	Integer 4	Código de incremento.
Nombresalumno	Character 25	Nombres del alumno
Apellidosalumno	Character 25	Apellidos del alumno
Área	Character 45	Área de estudio para reserva.

Reserva	Logic 1	Verifica si la reservación se efectúa.
---------	---------	--

Índices	Tipo
Id	Primary

Tabla Matricula.

Esta tabla contiene información de los alumnos que se matriculan durante un periodo lectivo.

Id	Integer 4	Código de incremento
IdAlum	Integer 4	Código de Alumno
IdArea	Integer 4	Código de área.
Fechaingreso	Date 8	Fecha de matricula.
Desde	Date 8	Fecha que inicia el periodo lectivo el alumno
Hasta	Date 8	Fecha de culminación del año lectivo del alumno.

Índices	Tipo
Id	Primary
Idalum	regular
Idarea	regular

Tabla detallematricula

Esta tabla contiene información del detalle de una matricula.

Id	Integer 4	Código de incremento
Idmatricula	Integer 4	Código de matricula
TotalPago	Numeric 4,2	Valor de una matricula

Índices

Tipo

Id	primary
Idmatricul	regular

Tabla pensiones

Esta tabla contiene la información del pago mensual que el representante cancela por un servicio.

Id	Integer 4	Código de incremento
Idalum	Integer 4	Código del alumno
Desde	Date 8	Mes que cancela
Hasta	Date 8	Hasta que mes cancela.
Valor	Numeric 4,2	Precio de la mensualidad.

Índice

tipo

Id	primary
IdAlum	regular

Tabla uniformes.

Esta tabla contiene la información detallada de los uniformes, como talla valor, cantidad.

Id	Integer 4	Código de incremento
Cantidad	Integer 4	Stock de uniformes.
Detalle	Character 90	Información detalla de los uniformes.
Talla	Numeric 4	Tipos de talla de los uniformes.
Subtotal	Numeric 4,2	Precio parcial por el uniforme.
Iva	Integer 4	Impuesto al valor agregado.
Total	Numeric 4,2	Total del precio a cancelar.

Índice**tipo**

Id primary

Tabla asignaProfesorarea.

Esta tabla contiene información de que profesor va a dictar clases en una determinada área.

Id	Integer 4	Código de incremento
Idprof	Integer 4	Código del profesor
Idarea	Integer 4	Código del área
Desde	Date 8	Periodo que empieza dar clases el profesor.
Hasta	Date 8	Periodo de finalización de clases del profesor en una determinada área..

Índice**tipo**

Id primary

Idprof regular

Idarea regular.

Tabla materias.

Esta tabla contiene la información de las materias que se dictan en una determinada área de estudio.

Id	Integer 4	Código de incremento
Idarea	Integer 4	Código de área.
Nombre	Character 30	Nombre de la materia.

Índice	Tipo
Id	Primary
Idarea	Regular.

Tabla Lunch

Esta tabla contiene información de alumnos que reciben este servicio, que ofrece la institución.

Id	Integer 4	Código de incremento
Idalumno	Integer 4	Código del alumno
Desde	Date 8	Fecha inicial del pago por el servicio
Hasta	Date 8	Fecha hasta cuando se realizó el pago por el servicio.
Impuesto	Float 4,2	Impuesto al valor agregado
Valor	Float 4,2	Total a pagar

Índice	tipo
Id	primary
Idalumno	regular.

Tabla Notas1

Esta tabla contiene información de las notas que tienen los alumnos.

MS = muy satisfactorio.

S = satisfactorio.

PS = poco satisfactorio.

Las notas a partir de segunda año de educación básica serán calificadas sobre 20.

Id	Integer 4	Código de incremento.
Idmateria	Integer 4	Código de materia.
Idalumno	Integer 4	Código de alumno
first_trim	Character 2	Nota del primer trimestre
sec_trim	Character 2	Nota del segundo trimestre.
ter_trim	Character 2	Nota del tercer trimestre
promtotal	Character 2	Sumatoria total de los 3 trimestres.

Índices	tipo
Id	primary
Idmateria	regular
Idalumno	regular

CAPITULO V

DISEÑO DEL SISTEMA.

5.1 Asignación de comportamientos de las clases encontradas.

5.2 Construcción del diagrama de secuencia.

Un diagrama de secuencia muestra la interacción de los objetos ordenados en secuencia temporal. Muestra los objetos que se encuentran en el escenario y la secuencia de mensajes intercambiados entre los objetos para llevar a cabo la funcionalidad descrita por el escenario.

A continuación se describe de manera grafica los diagramas de secuencia del sistema definidos para cada use case del sistema.

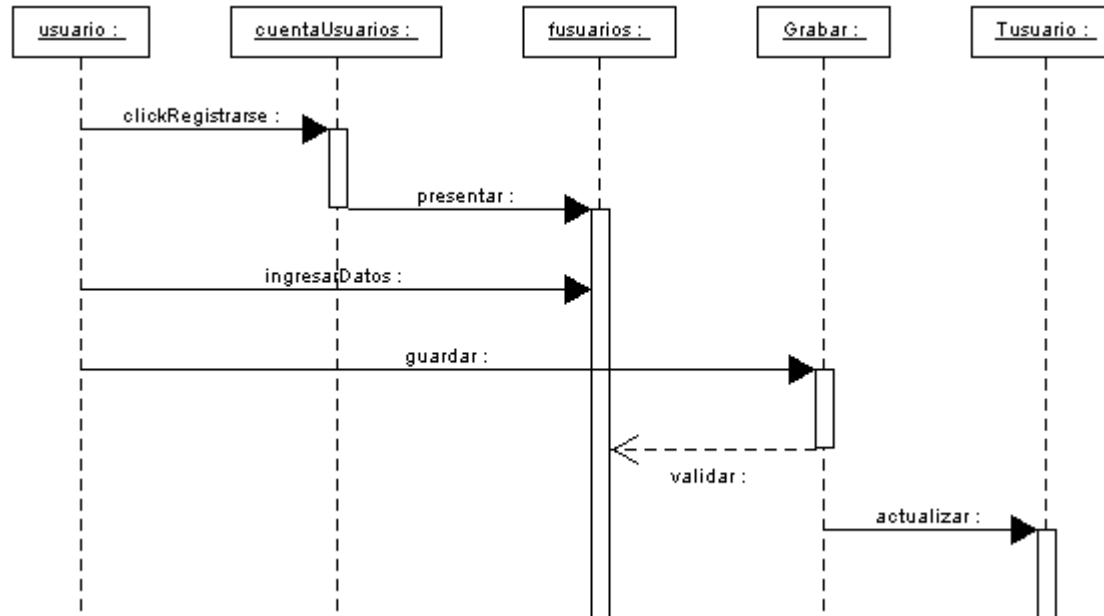
Diagrama de secuencia: Registrar Usuario.

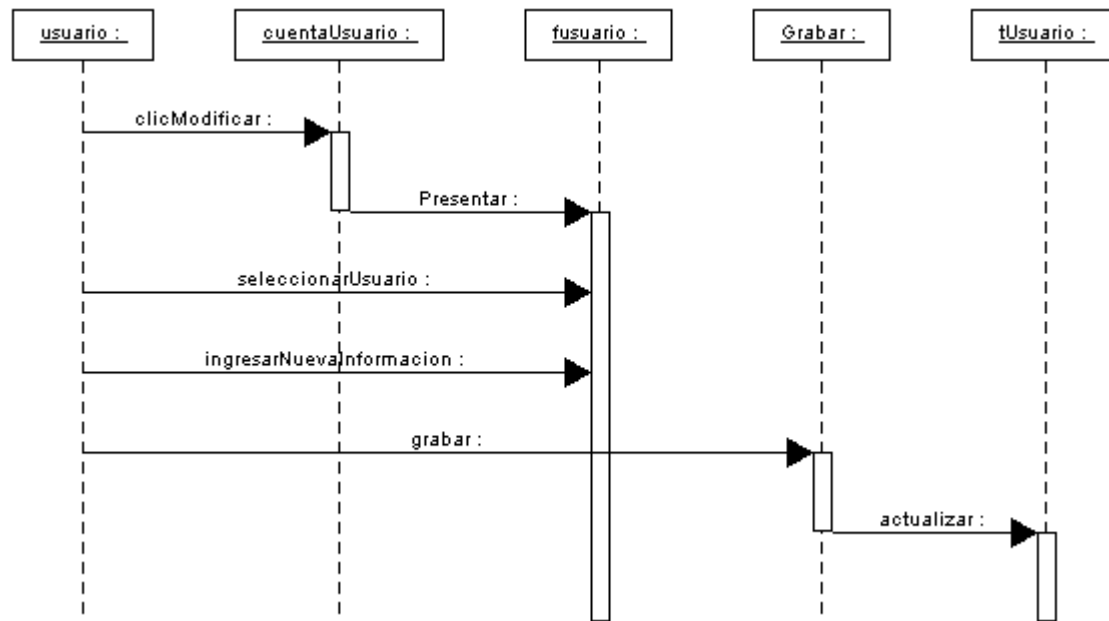
Diagrama de Secuencia: Modificar Usuario.

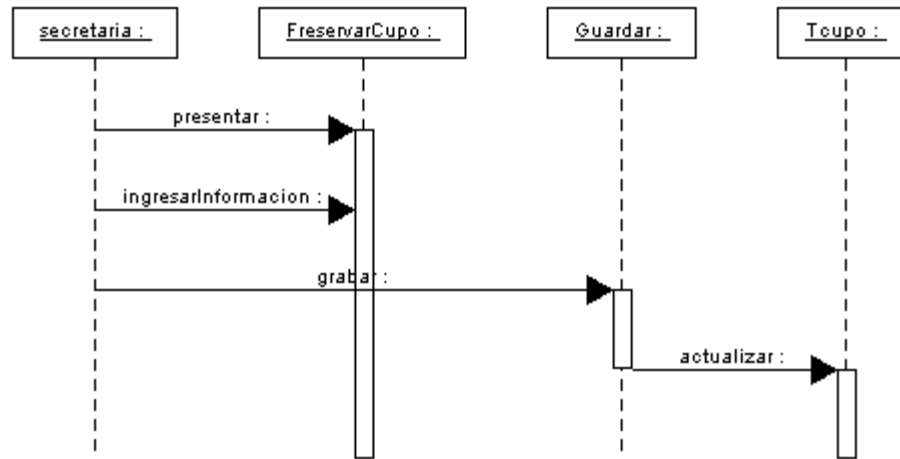
Diagrama de Secuencia. Reservar cupo.

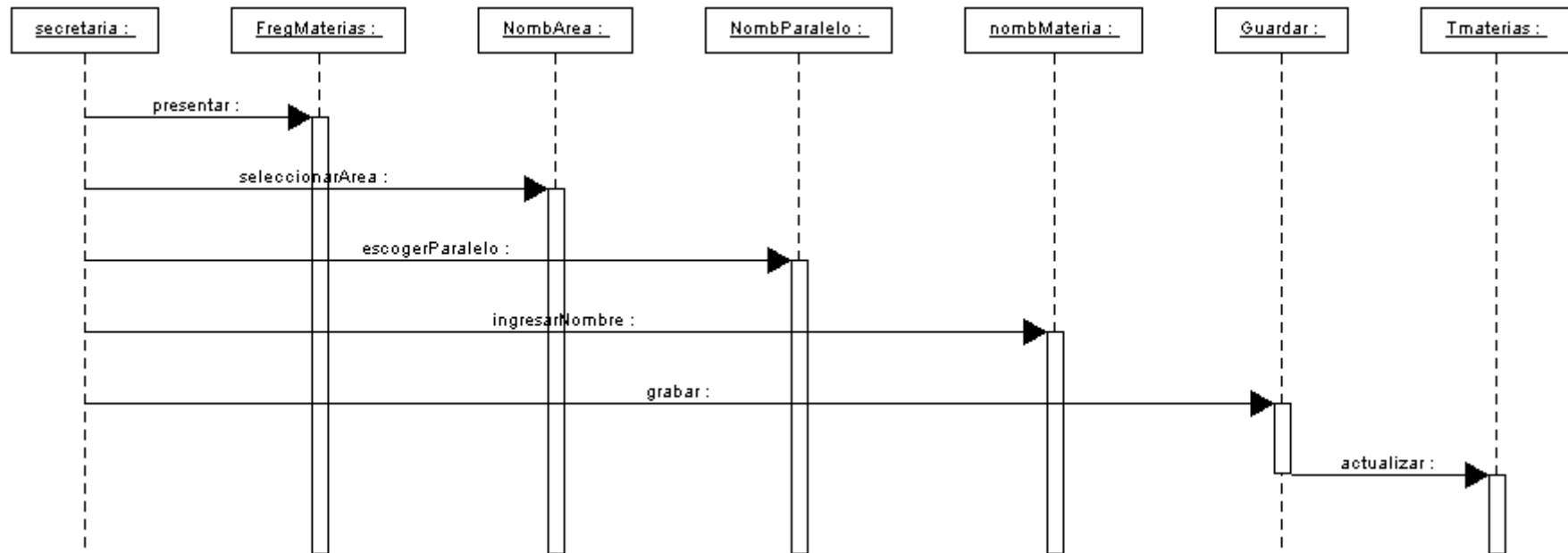
Diagrama de Secuencia: Registrar Materias.

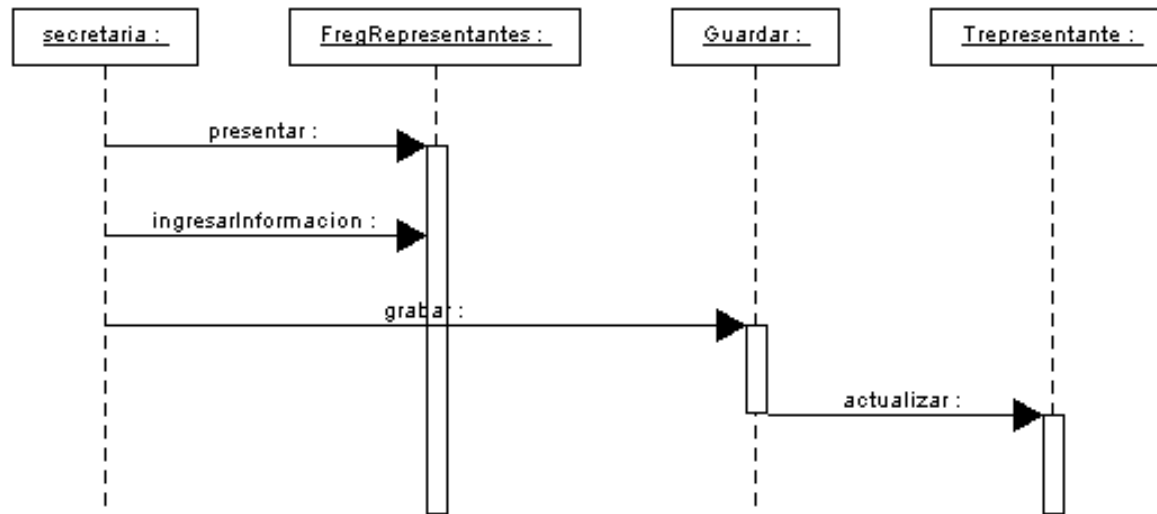
Diagrama de Secuencia: Registrar representante.

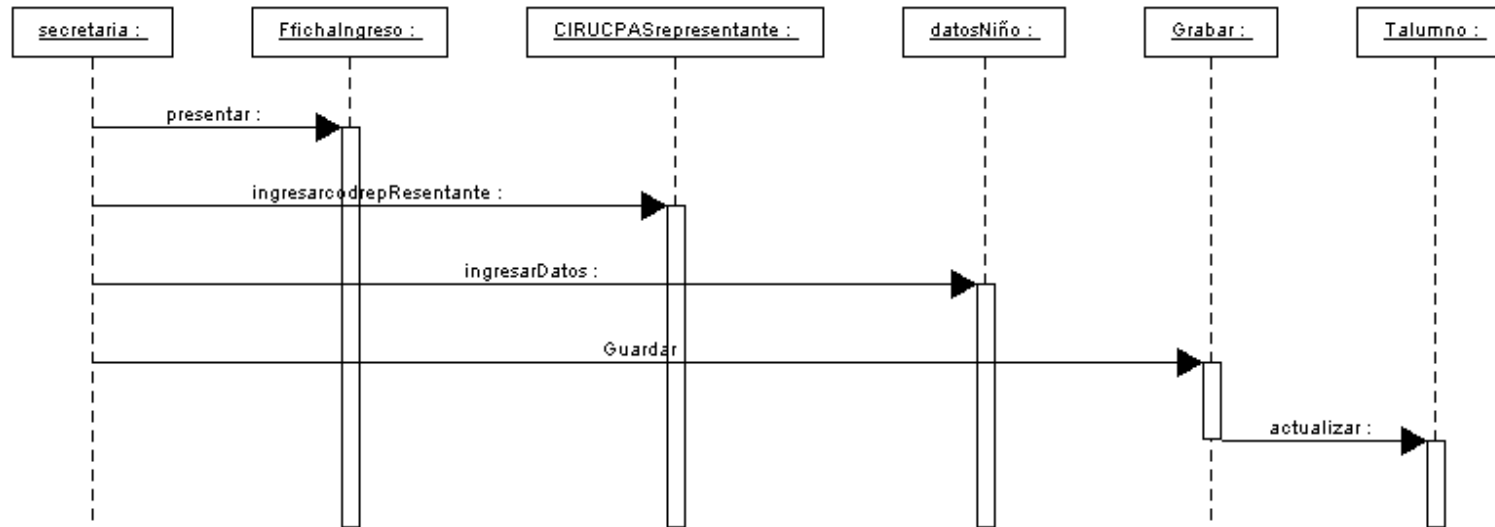
Diagrama de Secuencia: Ficha Ingreso.

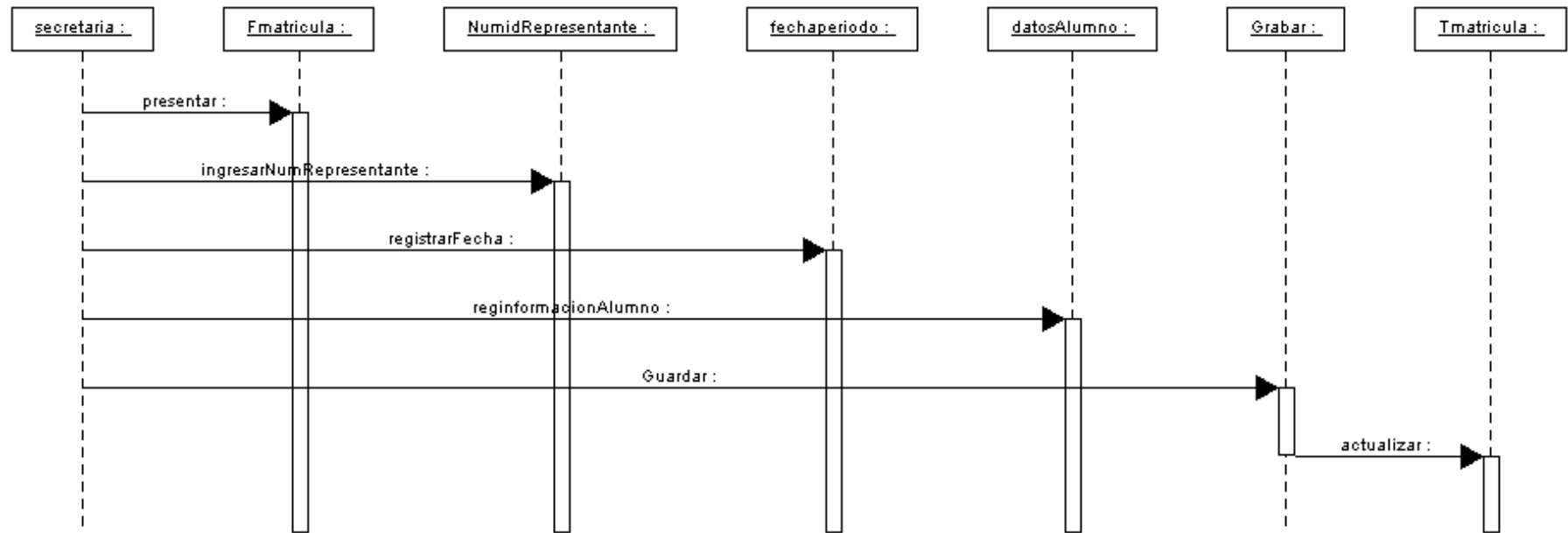
Diagrama de Secuencia: Registrar Matricula.

Diagrama de Secuencia: Modificar datos Alumno.

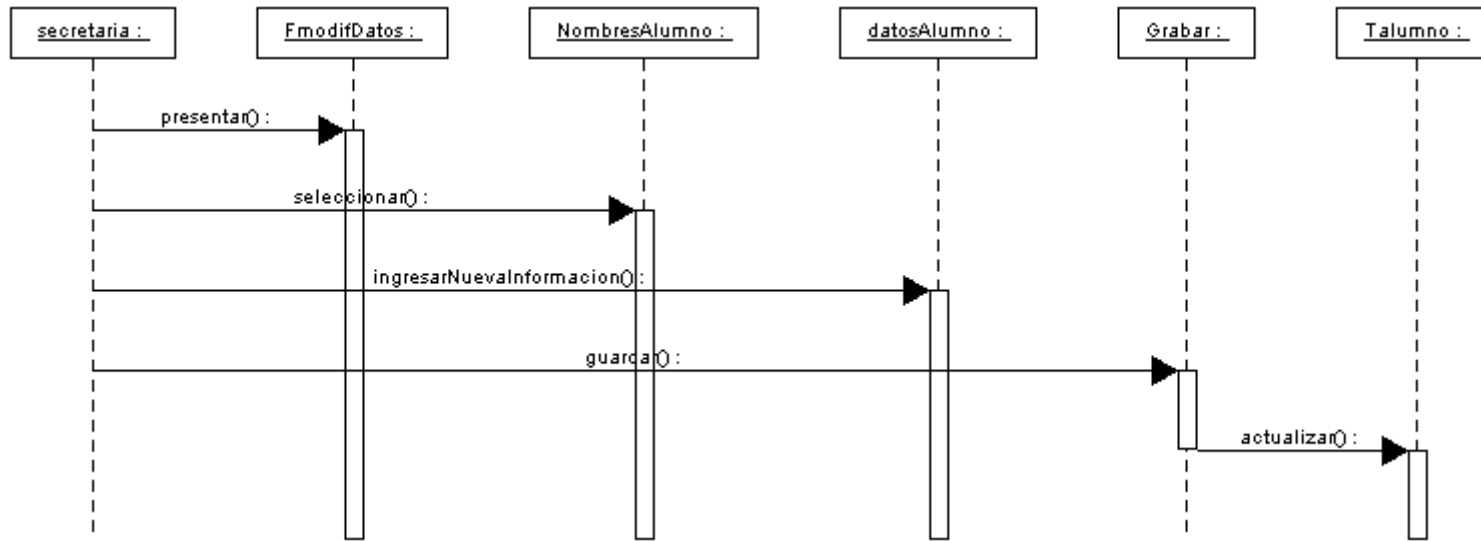


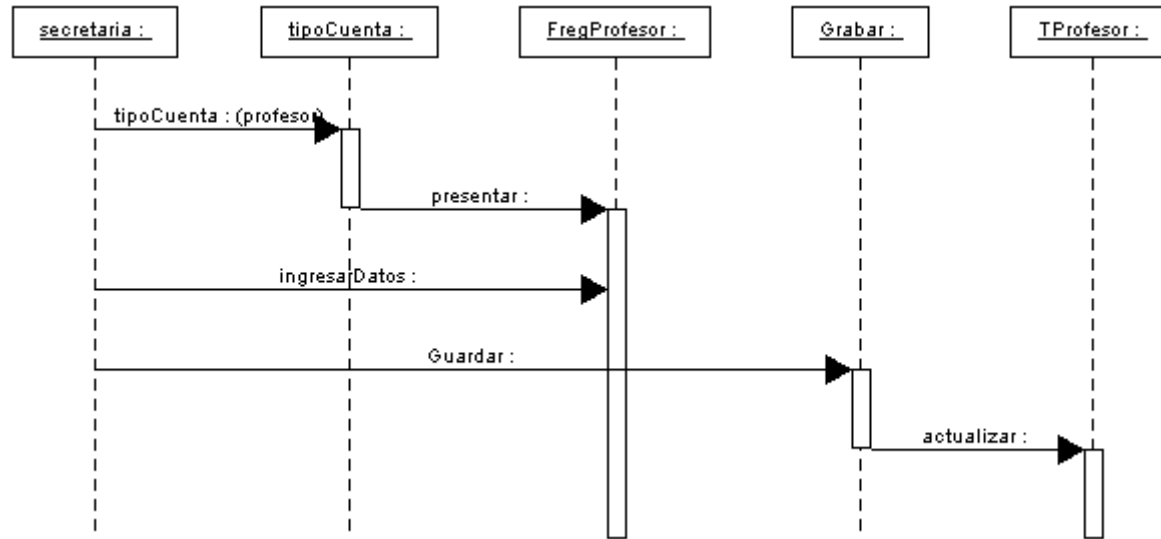
Diagrama de Secuencia: Registrar Profesor

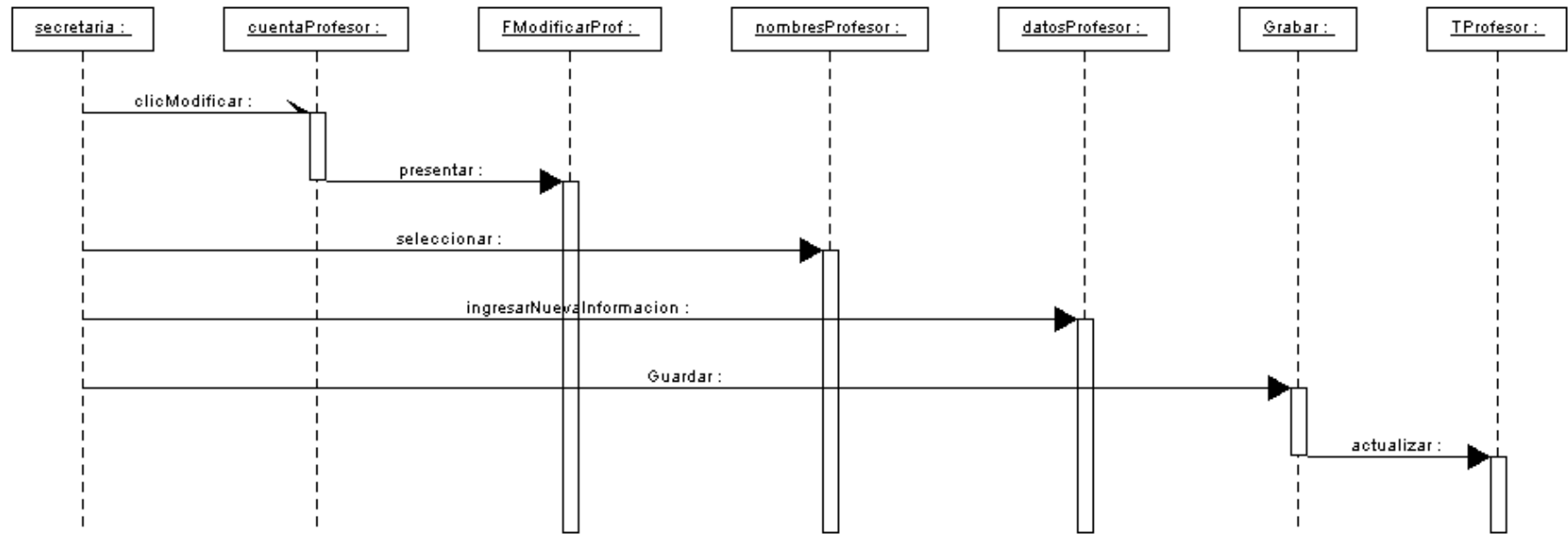
Diagrama de Secuencia: Modificar datos profesor.

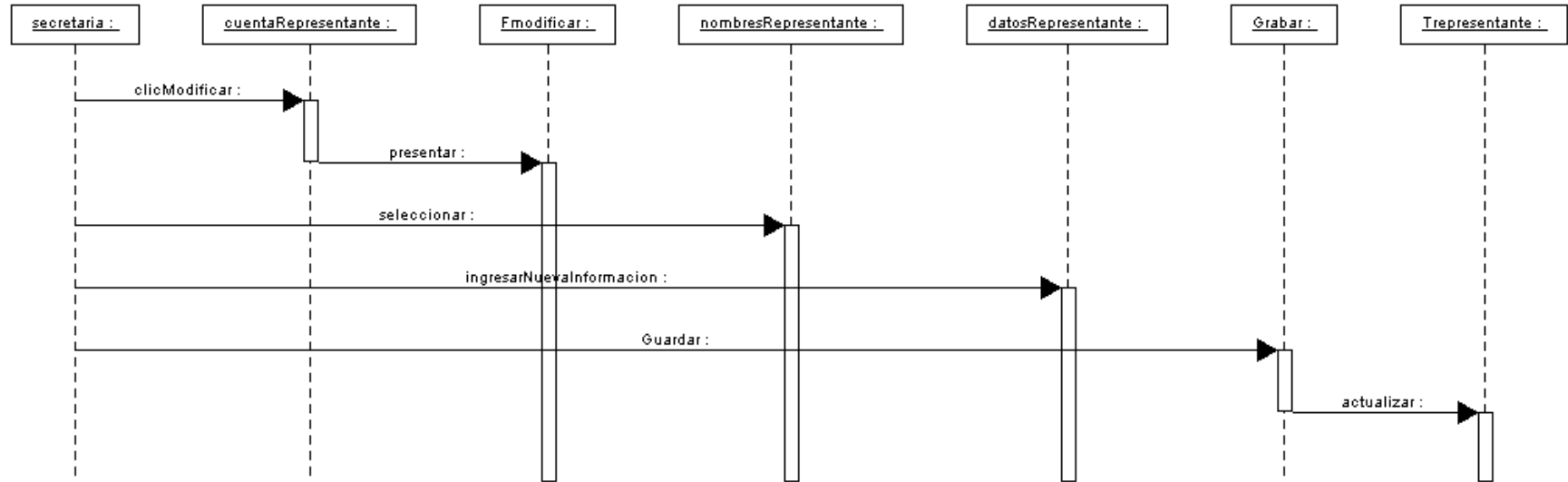
Diagrama de Secuencia: Modificar datos representante.

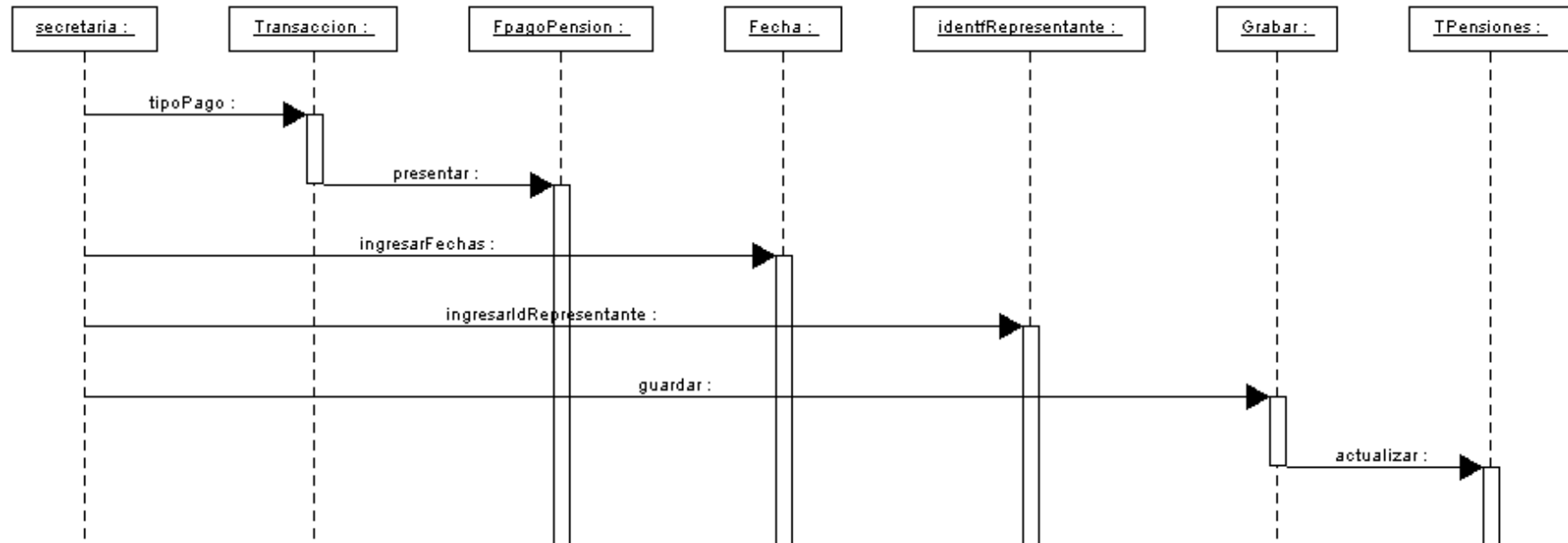
Diagrama de Secuencia: Registrar Pago de Pensiones.

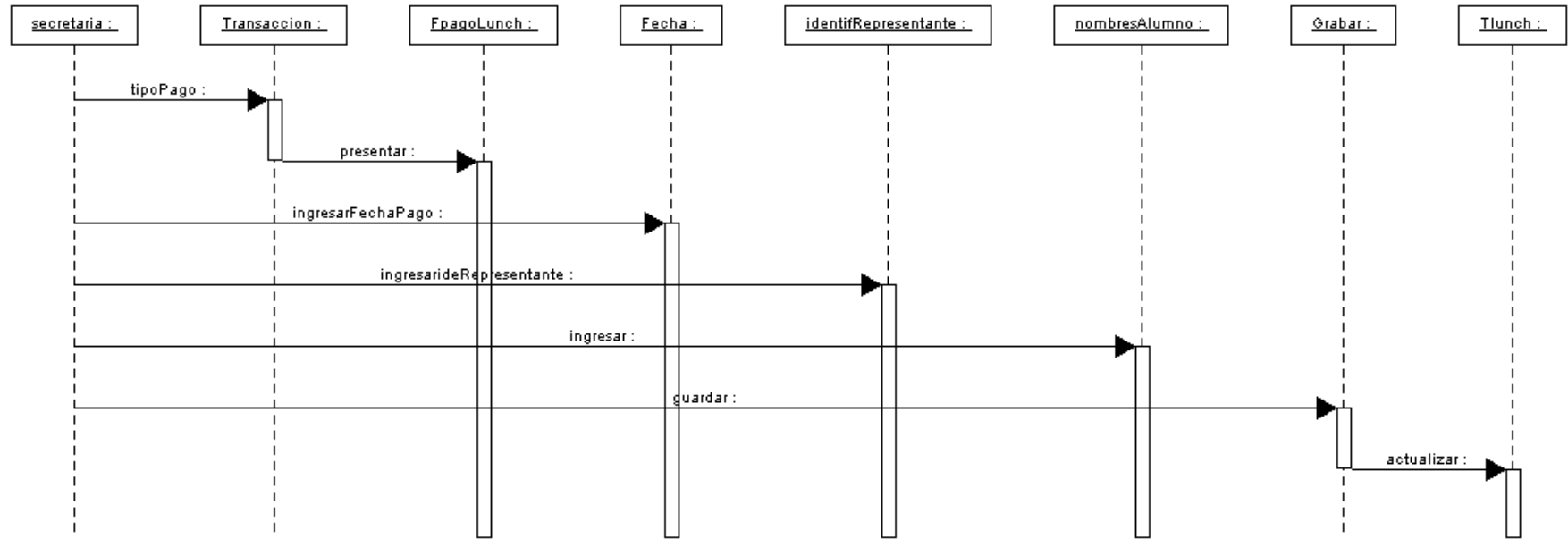
Diagrama de Secuencia: Registrar pago de Lunch.

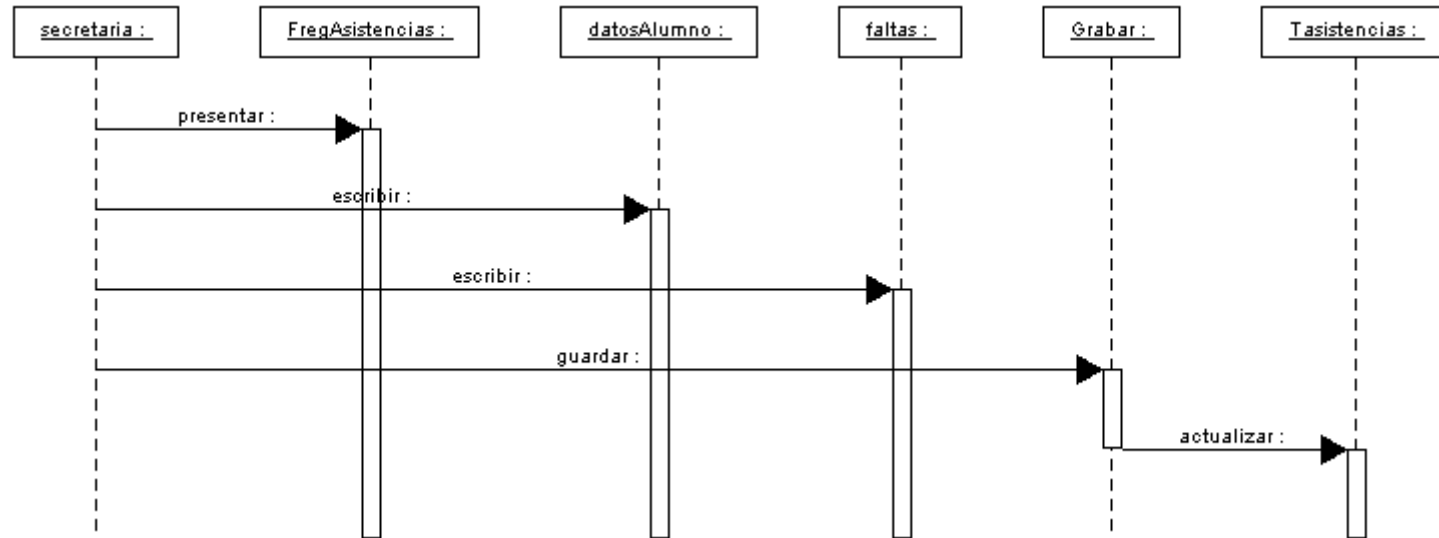
Diagrama de Secuencia: Registrar asistencias.

Diagrama de Secuencia: Registrar Calificaciones.

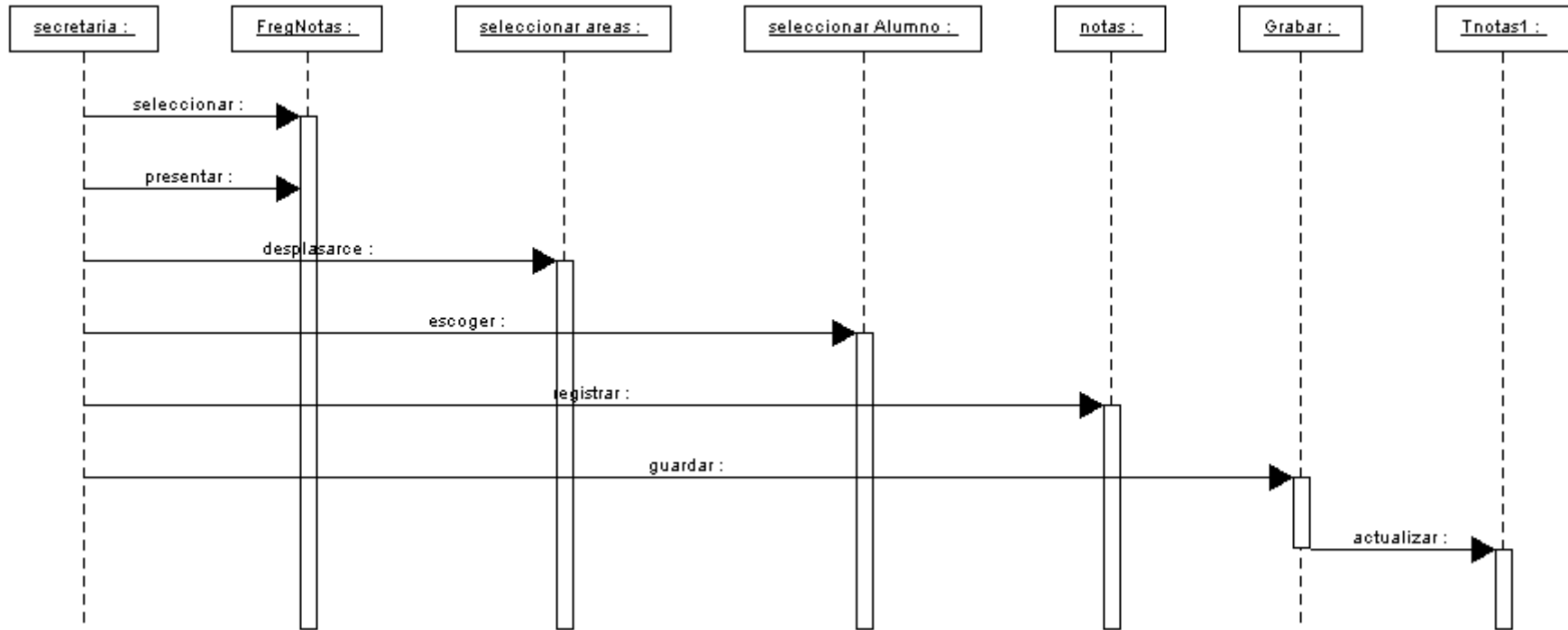


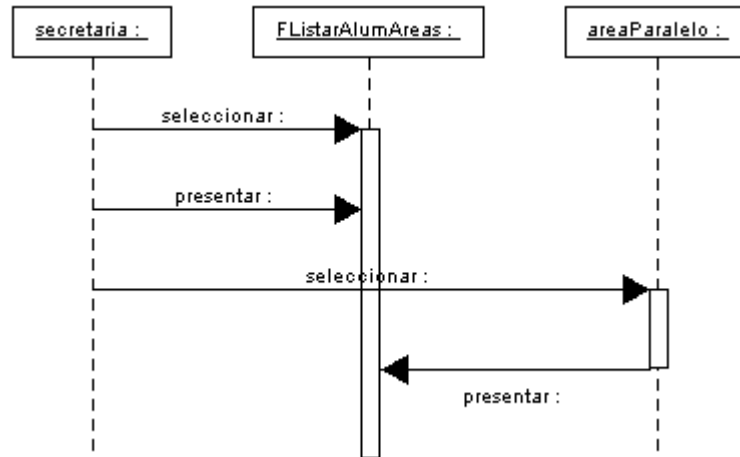
Diagrama de Secuencia: Listar alumnos por área.

Diagrama de Secuencia: Registrar Uniformes.

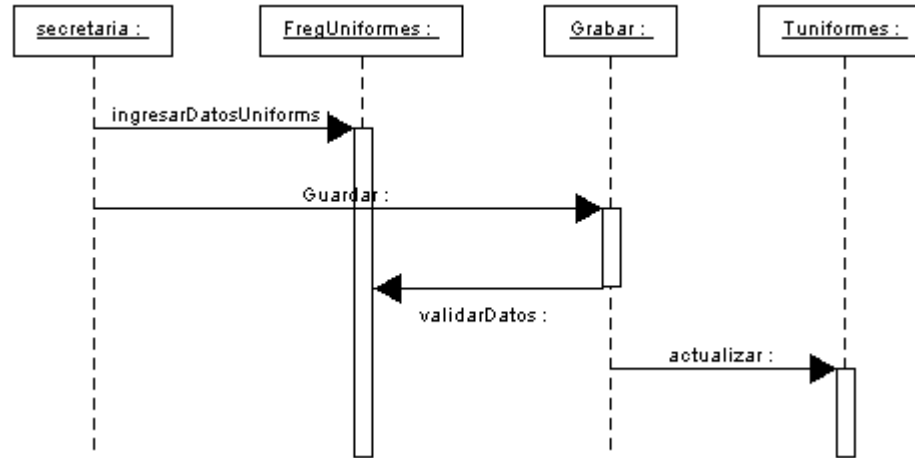


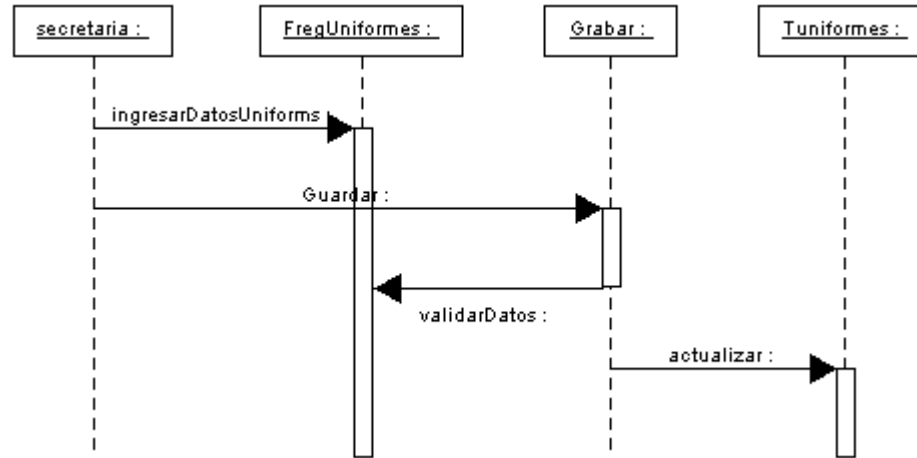
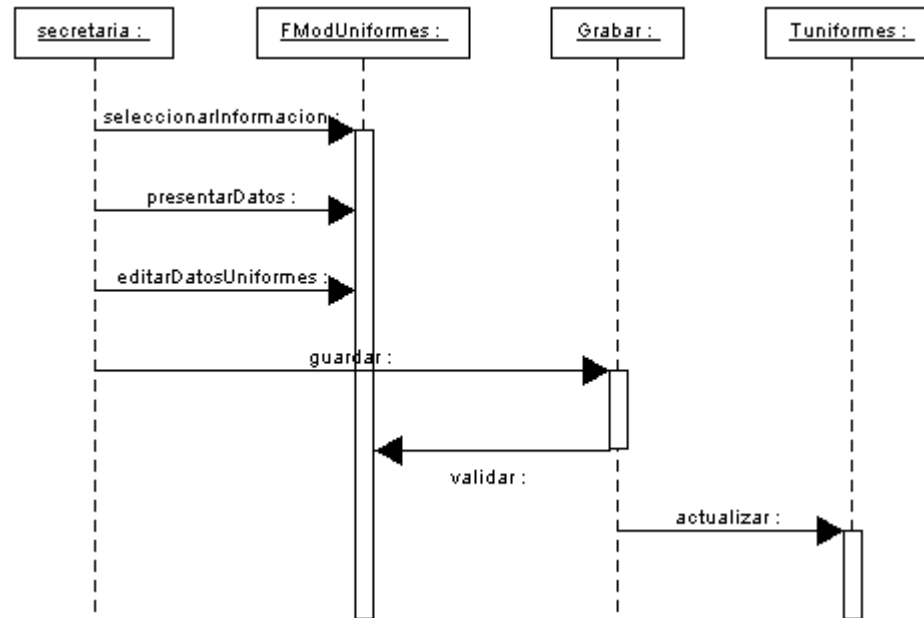
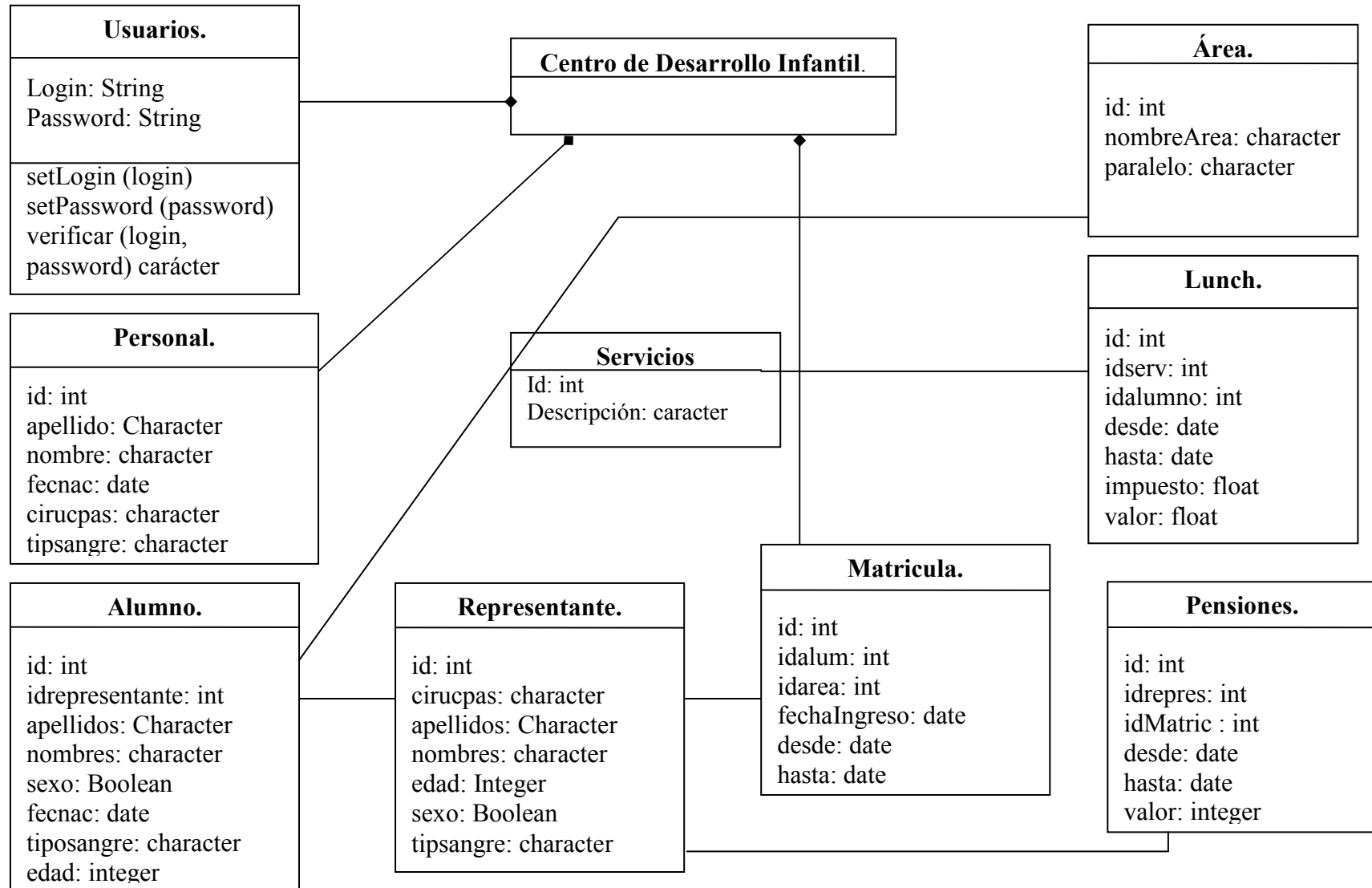
Diagrama de Secuencia: Venta de Uniformes.

Diagrama de Secuencia: Modificar datos de uniformes.

5.3 Actualización final del modelo Estático (Diagrama de Clases).



CAPITULO VI

IMPLEMENTACIÓN

DEL SISTEMA

6.1 Construir la Base de Datos a partir del diagrama de clases.

Definimos las tablas utilizarse y sus respectivos campos con el nombre (campo), tipo y el tamaño en bytes.

En la construcción de la Base de Datos se implementara, procedimientos almacenados con la finalidad de que se actualicen todas las tablas existentes y que tengan relación entre las mismas. A continuación se detallara el significado de estas.

Para la actualización de las tablas serán en cascada, se utiliza esta regla para que las tablas que tengan una relación de varios a muchos o viceversa se actualicen con sus respectivos índices; esto significa que no exista información incongruente. Especifica que los cambios en archivos de claves primarias o candidatas de una referida tabla se cambien automáticamente a los valores correspondientes, y se almacenen.

6.1.1 Selección del lenguaje y tecnología.

El lenguaje de programación tendrá impacto en la planificación, el análisis, el diseño, la codificación, las pruebas y el mantenimiento de un proyecto. El papel del lenguaje de programación ha de tenerse presente en todo momento; los lenguajes de programación los medios de la traducción hombre – maquina, sin embargo, la calidad del resultado final se encuentra mas fuertemente unida a las actividades de ingeniería de software que proceden y siguen a la codificación.

Una vez que se han establecido los requisitos del software las características del lenguaje de programación candidatos se hacen

más importantes. Así por ejemplo si se requieren complejas estructuras de datos, habrá que evaluar cuidadosamente los lenguajes de programación que soporten sofisticadas estructuras de datos. Si lo importante es un alto rendimiento y posibilidades de tiempo real, se debe especificar un lenguaje diseñado para aplicaciones de tiempo real o para eficiencia en memoria – velocidad. Se encuentran algunos lenguajes adecuados.

Idealmente los requisitos del software habrían de precipitar la elección de un lenguaje que mejor se ajuste al procesamiento que se ha de llevar a cabo.

La calidad del diseño de software viene dada por su independencia de las características de los lenguajes de programación. Sin embargo los atributos del lenguaje juegan, de hecho su papel en la calidad de un diseño acabado y afectan (tanto consciente como inconscientemente) a la forma de especificar el diseño.

Es así en mi caso se ha escogido el lenguaje de programación Visual Fox Pro 8.0, por sus características se acoge a las necesidades del diseño, permitiendo desarrollar una interfaz amigable, procesos seguros, y nos permite implementar los modelos de análisis y diseño creados mediante el Análisis Orientado a Objetos y Diseño Orientado a Objetos, Windows XP como sistema operativo, Gestor de Base de datos de Visual Fox Pro 8.0 para Windows, para el análisis y diseño se ha empleado la metodología Orientada a Objetos, y las técnicas de UML.

6.2 Escribir y generar código del sistema.

En esta etapa voy a describir la implementación de los métodos contenidos en los modelos de clases, describiendo los atributos y operaciones que contiene cada método representado en el modelo de clases del sistema

PROGRAMAS.

Main: Este programa es el que permite arrancar la aplicación, este se encarga de atribuir todos los PATHS del sistema, se hace el llamado a otros programas que contienen mas parámetros para que el sistema se comporte mas adecuadamente y así sea mas fácil su utilización, se definen algunos SET como SET EXCLUSIVE OFF, para que las tablas no se habrán en modo exclusivo, entre otros. Además de esto se colocara la ruta para que habrá la Base de datos, encuentre imágenes, etc. se ejecutaran el menú de inicio del sistema y el formulario principal el cual pedirá al usuario su login y password para acceder al sistema. A continuación el código.

```
CLEAR
```

```
CLEAR ALL
```

```
CLOSE DATABASES ALL
```

```
**** Variable que bloquea el menu
```

```
PUBLIC salto
```

```
salto = .T.
```

```
PUBLIC cdir
```

```
cAPP = SYS(16)
```

```
cdir = JUSTPATH(cApp)
```

```
*!* Configuración inicial de directorios y rutas de acceso
```

DO Progset.prg

DO comportamiento.prg

*** Programa que indicara al usuario que debe sacar respaldo de su Base de Datos.

DO recordatorio.prg

*** Formulario que valida el login y password del usuario.

DO FORM ingreso_sistema.scx

*** Ejecución del menu

DO menumain.mpr

READ EVENTS

ON ERROR

ON ERROR

*** Manejo de errors.

FUNCTION ManejaError(nError,nMessage)

 WAIT windows nMessage

ENDFUNC

CLASES

Componentes. Se encuentran todos componentes para la contracción del diseño grafico de los formularios, además código que será reutilizado. Tenemos: cuadrícula, formulario, grupobotones, imagen, jbutton, jcheck, jcombo, Judit, jlab, jlabel, jlist, joption en este se presentara el tipo de identificación del representante / profesor, jpage, jshape, jtext, jtimer, salir llevará el formulario, sx sexo de la persona.

AddMaterias. Esta clase contiene los métodos de guardar, nueva materia, nueva área y nuevo paralelo. En cada uno de estos métodos se hace referencia a guardar las nuevas materias, que pertenecen a las diferentes áreas de estudio.

Nuevo.click(). En este evento se realizara el llamado al método a.nuevo() el cual limpiara todos los campos para registrar una nueva área, paralelo con sus respectivas materias. Para limpiar el campo materia se dará doble clic sobre el campo Materia, en el cual se llamara al método a. newmatter(), y este solo limpiara el campo materia; para registrar una nueva materia.

addUniformes(). En esta clase se implementara el código que servirá para registrar uniformes de la institución; también se registrara la talla de los mismos, valor por pieza, IVA y el total a pagar.

Calcular.click(). En este botón se hará el llamado al método que realiza el cálculo automáticamente del total a pagar por el conjunto.

Guardar.click(). Llamara al método que guarda la información nueva.

Procedimiento a.buscar(). En este procedimiento se realizara la búsqueda para saber si los datos del uniforme están registrados, si están registrados se presentara un mensaje de que esos datos ya existen y si quiere agregar un nuevo stock.

asignarprofesor()

En esta clase se escribirá el código que permita, a un determinado profesor impartir clases a una determinada área de estudio. Se escogerá el área, nombres del profesor; se escribirá el periodo de inicio y final al cual dará clases.

guardar.click(). Hará el llamado al método el cual se encargara si ese periodo esta o no ya ocupado por un profesor; de no estar ningún maestro asignado guardara la información.

Nuevo.click(). Limpia los campos para asignar un nuevo maestro.

Asistencias(). Registrara las asistencias de alumnos, pertenecientes a una determinada área, y paralelo.

Guardar.click(). Aquí se realiza el llamado al método para validar, buscar, guardar las inasistencias del alumno.

Si el alumno tiene mas inasistencias se podrá realizar doble clic sobre el campo faltas para registrar una nueva fecha.

Nuevo.click(). Llamara al método para limpiar todos los campos, y así podrá registrar nuevos datos.

Freserva_cupo(). Presentara un listado de los alumnos que han reservado un cupo, en las diferentes áreas.

Agregar.click(). Hará el llamado a un formulario para registrar los datos del alumno, así como al área al cual desea reservar el cupo.

Borrar.click(). Borrara lógicamente los datos de una reservación, previamente seleccionada.

Restaurartodo.cliclk(). Reintegrara toda la información que fue eliminada previamente.

Classreservarcupo(). Presentara un formulario en el cual se podrá realizar la reservación de un cupo.

Guardar.click(). Llamara al método reservacion1.Guardar(), el cual permitirá registrar la nueva información.

Nuevo.click(). Limpiara los campos para registrar nueva información.

Imprimir.click(). Aquí llamara al método `.reservacion1.Imprimir()` el cual presentara un reporte del cupo, con los respectivos datos.

Cmoduniformes.vcx. Presentara un formulario el cual mostrara la información de los uniformes; así como botones de navegación, los cuales permitirán desplazarse por los diferentes datos de los uniformes.

Guardar.click(). Permitirá almacenar los cambios realizados.

Fichaingreso(). Presentara un formulario el cual permitirá registrar la información mas detallada del infante.

Joption1.interactiveChange(). Implementara la longitud del numero de cedula, ruc o pasaporte del representante del niño.

Guardar.click(). Llamara al método `fi.guardar()`. El cual permitirá almacenar la información del niño.

Nuevo.click(). Limpiara todos los campos, ficha de ingreso para registrar una nueva hoja de ingreso.

Imprimir.click(). Llamara al método `Fi1.Imprimir()`, el cual permitirá mostrar un reporte, para ser impreso; con los datos actuales.

Listados(). Presentara un listado de los alumnos que están matriculados

Buscar.click(). Permitirá realizar la búsqueda de los alumnos matriculados en una determinada área.

Imprimr.click(). Presentara un reporte el cual contendrá el área de estudio, paralelo, nombres del profesor; y los alumnos pertenecientes a dicha área.

Lunch(). Presentara un formulario el cual permitirá registrar los datos de alumnos que se apuntaron para recibir el servicio de lunch.

Calcular.click(). Llamara al método Clase1.calcular() para calcular el costo a pagar por el representante, para que su representado reciba el servicio de lunch.

Guardar.click(). Llamara al método clase1.Guardar(), así se podrá almacenar los datos del niño, costo, numero de identificación del representante, de los meses que ha cancelado para recibir dicho servicio.

Imprimir.click(). Presentara un reporte el cual contiene la información de los meses cancelados, identificación del padre, nombres del niño, IVA, total a pagar.

Nuevo.click(). Llamara al método clase1.Nuevo(), para limpiar todos los campos.

Matricula(). Presentara un formulario, el cual permitirá registrar una matricula.

Guardar.click(). Llara al método metodosmat1.Guardar(), el cual validara, buscara y permitirá registrar los datos de los alumnos, periodo, área y el valor a pagar.

Nuevo.click(). Llamara al método metodosmat1.Nuevo(), que permitirá limpiar los campos para registrar una nueva matricula.

Imprimir.click(). Llama al procedimiento Metodosmat1.imprimir(), para presentar un reporte el cual constara de: identificación del representante, datos del alumno, nivel en el que se matricula, costo a pagar y el periodo lectivo.

Modifuser(). Presentara un formulario, para modificar los datos de los usuarios que tienen acceso al sistema.

Buscar.click(). Llamara al método A1.buscar(), el cual validara y buscara los datos a modificar. Una vez que la búsqueda ha tenido éxito hará visible el campo a con los datos a modificar.

Guardar.click(). Permitirá llamar al método A1.Guardar(), aquí se realizara la convalidación del usuario, a modificar y guardara la nueva información.

Nuevo.click(). Hará el llamado al método A1.nuevo(), el cual limpiara y hará no visible los campos nuevos datos.

Pensiones(). Presentara un formulario el cual permitirá registrar el pago de pensiones así como la fecha, descuento, nombres del alumno y la identificación del representante.

Guardar.click(). Llamara al método B1.Guardar(), el cual buscara, validara los campos, calculara el valor a pagar y guardara los datos en la respectiva tabla.

Nuevo.click(). Limpiara los campos para registrar un nuevo pago de pensiones.

Imprimir.click(). Llamara al método B1.Imprimir(); el cual presentara un reporte con los datos que se almacenaron.

Registroáreas(). Presentara un formulario en el cual se podrá registrar las áreas de estudio así como el paralelo, para registrar un paralelo, con distinto nombre, se realizara doble clic sobre el campo paralelo.

Guardar.click(). En este evento se hará el llamado al método codigo1.Guardar() en el cual se buscara, validara y registrara la información de las nueva áreas de estudio en sus respectivas tablas.

Nuevo.click(). Llamara al método `codigo1.nuevo()`, este limpiara todos los campos para registrar una nueva área de estudio.

Regpersonal(). Presentara un formulario el cual permitirá registrar los datos del personal que labora en el Centro de Desarrollo Infantil.

Guardar.click(). Llamara al método `a1.guardar()` el cual validara, buscara, si esta registrado o no; y almacenara la nueva información de los profesores.

Nuevo.click(). Llamara al método `a1.Nuevo()`, que limpiara todos los campos para registrar nuevos datos.

Regrepresentante(). Presentara un formulario en el cual se podrá registrar los datos de los representantes.

Guardar.clcik(). Llamara al método `C1.Guardar()` el cual validara, buscara y registrara los datos del representante.

Joption.interactiveChange(). Implementara la longitud del numero de cedula, ruc o pasaporte del representante.

Nuevo.click(). Llamara al método `C1.Nuevo()`, que limpiara todos los campos para registrar datos de un nuevo representante.

regUser(). Presentara un formulario en el cual se podrá registrar nuevos usuarios que tendrán acceso al sistema.

Guardar.click(). Llamara al método `C1.Guardar()`, que validara, buscara y registrara los datos del nuevo usuario.

Nuevo.click().llamara al método `C1.Nuevo()` , que limpiara los campos para registrar un nuevo usuario.

Saleuniforms(). Permitirá registra la venta de uniformes de la institución, para alumnos.

Calcular.click(). Llamara al método B1.Calcular(), para realizar el calculo automático del total a pagar, por la compra de uniformes.

Guardar.click(). Llamara al método B1.Guardar() en el cual validara, buscara, y registrar la información de la venta de uniformes.

Nuevo.click(). Aquí se realizara el llamado al método b1.Nuevo(), que limpiara todos los campos.

Imprimir.click(). Llamara al método b1.IMprimir() que presentara un reporte con los datos de adquisición de uniformes, (identificación del representante, descripción del uniforme, nombres del alumno, subtotal, IVA, descuento y total a pagar).

stockUniforms(). Presentara un listado completo de los uniformes; existntes.

Usuarios(). Presentara el formulario de acceso al sistema.

Aceptar.click(). Llamara al método A1.Buscar(), que buscara si el usuario así como la contraseña están almacenadas en la Base de Datos.

Cancelar.click(). Llamara al programa de salida.

respaldar_datos(). Este formulario permitirá sacar respaldos de la Base de Datos, con la finalidad de: empezar nuevos periodos, posibles ataques de virus, etc. así la información no se perderá y se la tendrá siempre disponible.

Botón ... Mediante este botón se seleccionara la carpeta que se desea respaldar:

Nombre archivo. En este campo se colocara un nombre con el cual se identificara el archivo respaldado.

Crear Zip.click(). En este evento llamara primeramente validara los campos si están con la respectiva información, entonces llamara al método form1.zip(); y creara el respectivo apoyo.

6.3 Ejecutar pruebas de unidad e integración del Sistema.

Se debe tener en cuenta lo siguiente:

Para que la aplicación se instale en cualquier parte del disco duro, se construirá el siguiente código.

```
SET PATH TO(SYS(2004))
```

```
gDir = SYS(5) + SYS(2003) + "\BDD" + "," + SYS(2003) + "\Forms" + "," + ;
```

```
SYS(2003) + "\PRGS" + "," + SYS(2003) + "\Include" + "," + SYS(2003) +  
"\Graphics" + "," + ;
```

```
SYS(2003) + "\Informs" + "," + SYS(2003) + "\Menus" + "," + SYS(2003) +  
"\LIBS"
```

```
SET PATH TO (gDir)
```

```
SET DEFAULT TO SYS(5) + SYS(2003)
```

```
*!* Apertura de la Base de datos
```

```
IF NOT USED('Tesis') THEN
```

```
OPEN DATABASE " tesis.dbc" EXCLUSIVE
```

ENDIF

Se crea un proyecto con cualquier nombre por ejemplo: Academic

Se tendrá que escribir código para la captura de errores, con la finalidad que no aparezcan mensajes propios de VFP, más bien los mensajes que se mostraran serán los que el programador asigne y aparezcan cuando se produzca un error.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones.

Una vez realizado el presente trabajo he llegado a las siguientes conclusiones:

- Es posible implementar el presente sistema académico.
- Los objetivos planteados se cumplieron en un 100%.
- Por medio de este sistema se podrá automatizar el centro de desarrollo Infantil “Angelitos de Luz”, con la finalidad de agilizar el proceso académico.
- Con la utilización del sistema se podrá lograr una contabilidad de los datos almacenados en los diferentes archivos de la Base de Datos.
- Con la utilización del sistema académico, se podrá mejorar la atención al público en general.
- Los objetivos planteados de analizar, diseñar e implementar un sistema académico y dejar un instrumento de consulta sobre análisis, diseño e implementación de sistemas.
- La metodología de desarrollo utilizada UML, brinda facilidades para el desarrollo de software a la medida.
- La parte más compleja de implementar el sistema es mantener la identificación con el usuario.

Recomendaciones.

Aunque no soy experto en el desarrollo de software a la medida doy las siguientes recomendaciones.

- Que se utilice la tecnología de desarrollo de sistemas informáticos como Visual Fox Pro como alternativas para dar soluciones informáticas a los problemas de administración de información.
- Siempre en el análisis se debe procurar sacar la mayor cantidad de información posible de la empresa o institución para la cual se desarrolla el sistema, para en lo posterior no tener problemas de no saber la manera en que funciona tal o cual proceso.
- Que la manipulación del sistema se lleve a cabo con disciplina, para lograr una mejor administración y control de los procesos a el encomendados.
- Al análisis y al diseño el analista debe concretar la mayor cantidad de tiempo, para así evitar complicaciones en el momento de la implementación.
- Realizar un diseño original y fácil de las pantallas, entretiene al usuario y aumenta la permanencia de la lectura e interactividad.
- Utilizar los estándares de desarrollo de software como lo es UML, el cual define patrones de diseño que entienda cualquier analista de sistemas.
- Contratar una persona especializada para que de mantenimiento al sistema.

- Sacar copias de la información registrada en el sistema académico por lo menos una vez por semana.

BIBLIOGRAFÍA.

<http://www.clikear.com/manuales/uml/faseconstruccionaltonivel.asp>

<http://dis.um.es/~jmolina/jis2000modeladonegocio.pdf>

<http://www.clikear.com/manuales/uml/diagramascasouso>.

<http://www.fpress.com/revista/Num0009/art.htm>

<http://www.portalfox.com/modules.php>

<http://usuarios.lycos.es/foxjose/importan/nuevo.htm>

<http://www.ita-software.com/foxpage.htm>

<http://www.davphantom.net/descarga.asp>

<http://www.fpress.com/revista/Num0009/art.htm>

<http://www.fpress.com/revista/Num1103/Truco.htm>

<http://www.rational.com/uml>

<http://www.monografias.com/trabajos/guiafox/guiafox.shtml>

<http://www.monografias.com/trabajos10/diusuar/diusuar.shtml>

<http://www.monografias.com/trabajos3/cursovfp/cursovfp.shtml>

<http://www.monografias.com/trabajos10/soft/soft.shtml>

<http://www.monografias.com/trabajos11/anasiste/anasiste.shtml>

<http://www.dcc.uchile.cl/~psalinas/uml/modelo.html>

<http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>

Comentarios.

jfpanchogp@gmail.com