

FACULTAD DE INGENIERIA

ESCUELA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS Y COMPUTACIÓN

TEMA:

DISEÑO, DESARROLLO E IMPLEMENTACION DEL SISTEMA ACADEMICO EN ESCUELA FISCAL MIXTA "TEODORO WOLF N°72" DE SAN LORENZO – MANTA - MANABÍ

AUTOR:

JOSE VICENTE MENDOZA MACIAS

QUITO, 2012

ÍNDICE

CAPITULO I: FUNDAMENTOS TEORICOS **4**

1.1 XTREME PROGRAMMING **4**

1.1.1 Introducción al Xtreme Programming (XP) **4**

1.1.2 Definición de Xtreme Programming (XP) 6

1.1.3 Actividades de XP 8

1.1.4 Valores del XP 9

1.1.5 Buenas prácticas del XP 10

1.1.6 Ciclo de vida del XP 13

1.1.7 Actores y responsabilidades del XP 15

1.1.8 Herramientas de análisis y diseño del XP 17

1.1.9 Aplicabilidad del XP 19

1.1.10 Críticas al XP 19

1.2 SOFTWARE LIBRE **21**

1.2.1 Introducción del Software Libre 21

1.2.2 Historia del Software Libre 21

1.2.3 Definición del Software Libre 23

1.2.4 Cuatro libertades del Software Libre 24

1.2.5 Licencias del Software Libre 25

1.2.6 Software Libre y el Código Abierto 27

1.2.7 Java 28

1.2.7.1 Historia de Java 28

1.2.7.1 Definición de Java 29

1.2.7.2 Característica de Java 30

1.2.7.2.1 Programación Orientada a Objetos 31

1.2.7.2.1.1 Elementos de POO 31

1.2.7.2.1.2 Características de POO 32

1.2.7.2.2 Independencia de Plataforma 33

1.2.7.2.3 Recolector de Basura 34

1.3 HERRAMIENTAS **34**

1.3.1 NetBeans 34
1.3.2 MySql 36
CAPITULO II: **37**
2.1 DESCRIPCIÓN DEL PROBLEMA **37**
2.2 OBJETIVOS ESTABLECIDOS **37**
2.3 LEVANTAMIENTO DE REQUERIMIENTOS **38**
2.4 ANÁLISIS DE REQUERIMIENTOS **41**
2.5 PROTOTIPO DEL SISTEMA ACADÉMICO **42**
2.6 PLANIFICACIÓN DE ENTREGAS **45**
CAPITULO III: DESARROLLO DEL SISTEMA ACADÉMICO **48**
3.1 ESTRUCTURA DE REQUERIMENTOS **48**
3.1.1 Diagramas de casos de uso a detalle 48
3.2 DISEÑO DE PROCESOS **95**
3.2.1 Diagramación de Procesos 95
3.2.2 Diagramación de Clases 103
3.2.3 Diagramas de secuencia 107
3.3 Pruebas del sistema **132**
CAPITULO IV: PRUEBAS E IMPLEMENTACIÓN **143**
4.1 Pruebas de usuarios **143**
4.2 Modificaciones finales **158**
4.3 Implementación del sistema **159**
4.3.1 Estándares de codificación 159
4.3.2 Código fuente 161
4.4 Mantenimiento del sistema **161**
CAPITULO V: CONCLUSIONES Y RECOMENDACIONES **162**
5.1 Conclusiones **162**
5.2 Recomendaciones **163**
BIBLIOGRAFIA **165**
ANEXO A **169**
ANEXO B **172**
ANEXO C **206**

CAPITULO I: FUNDAMENTOS TEORICOS

En este capítulo se van a explicar las bases teóricas mediante las cuales se van a realizar el proyecto. También se presentara una definición de las herramientas que son necesarios para el desarrollo del producto.

1.1 XTREME PROGRAMMING

1.1.1 Introducción al Xtreme Programming (XP)

Las notaciones de modelado y luego las herramientas fueron tomadas como la salvación para el éxito de desarrollo de software pero estas expectativas no fueron cumplidas, debido a que dejaba rezagada otro componente importante, las metodologías. Ya que no sirven de nada tener buenas notaciones y las herramientas necesarias sino se tiene directivas y normas en el desarrollo de software. Antes, la creación de aplicaciones solo se basaba en un esquema lleno de roles, actividades y documentación pero este esquema no se ajusta a la necesidades actuales donde el sistema es muy cambiante, los requerimientos no definidos en su totalidad y sobre todo el tiempo de desarrollo es reducido radicalmente, pero se requiere la misma calidad.

Debido a ésto es que comienzan a emerger las metodologías "ágiles", que están orientadas en su mayoría a pequeños proyectos y tienen como virtud manejar plazos reducidos, objetivos cambiantes y el uso de nuevas tecnologías. Estas nuevas metodologías han tomado tal importancia que existen conferencias internacionales donde tratan su adaptación al entorno y la diversidad de campos que se puede aplicar. En una conferencia realizada en febrero del 2001 en Utah-EEUU, por parte de 17 expertos de la industria de software es donde nace el término "ágil". Tenían como objetivo delinear los valores y reglas que deberían seguir los equipos de desarrollo a lo largo del proyecto, tratando de dar una alternativa a las metodologías tradicionales. Se creó el "Manifiesto Ágil" donde están los principales valores de este desarrollo:

- Al individuo y las interacciones del equipo de desarrollo, sobre el proceso y las herramientas

- Desarrollar software que funciona, más que conseguir una buena documentación
- La colaboración con el cliente, más que la negociación de un contrato
- Responder a los cambios, más que seguir estrictamente un plan

El Xtreme Programming (XP) es una metodología de desarrollo de software que pertenece a las "ágiles", debido a que se centra en el logro de metas, y no en las robustas y burócratas metodologías anteriores, toma como principal objetivo realizar un software de calidad y en el menor tiempo posible. El XP tiene como principal componente a los clientes, y no a los procesos, se basa en lanzamientos periódicos de pequeños ciclos de desarrollo con el fin de incluir puntos de control que puedan recibir cualquier nuevo requerimiento de los clientes para así mejorar la productividad y eficacia del software final.

La metodología XP surge en 1996 de la mano de Kent Beck, durante su trabajo en DaimlerChrysler, para el proyecto Chrysler Comprehensive Compensation System (C3). Beck había trabajado en el primer leguaje orientado a objetos que en verdad se hizo popular, el Smalltalk, y junto con Ward Cunninghan trataban de encontrar una forma de desarrollar software de una manera más simple y eficiente. Su trabajo recién vio la luz cuando en el C3 debido a las necesidades del proyecto, Beck aplica sus investigaciones, denominándolas "Xtreme Programming"

Para ayudar a perfeccionar el método Beck invito a Ron Jeffries, que se desempeño como entrenador del equipo de C3, inculcando las buenas prácticas del XP, también se realizaron algunos cambios sobre las prácticas por parte del colaborador habitual Ward Cunningham.

Beck ha editado varios libros sobre el XP, como por ejemplo "Xtreme Programming Explained" donde explica mucho más a detalle sus ideas. "Xtreme Programming" en su traducción al español "Programación Extrema" se refiere a que se debe seguir las buenas prácticas de manera constante pero a un nivel extremo.

1.1.2 Definición de Xtreme Programming (XP)

El Xtreme Programming es una metodología ágil que se basa en fomentar las relaciones interpersonales como fundamento, para llegar al éxito de desarrollo de software, promueve el trabajo en equipo, a su vez capacitando y estableciendo un buen clima laboral. El XP busca dos objetivos claramente establecidos que son realizar un software de calidad y de la forma más rápida posible (se debe tomar en cuenta que estos objetivos son los fundamentales de cualquier metodología ágil o tradicional). La forma de realizarlo es con una retroalimentación continua con el cliente, comunicación fluida con todos los que conforman el equipo del proyecto (usuarios, administrativos, etc.), simplicidad en las soluciones que se desarrollan y aptitud para enfrentar los cambios ya que esta metodología se la utiliza en proyectos donde los requerimientos son cambiantes o incompletos por lo que existe un alto riesgo técnico. Aún con estas metodologías actuales, un gran porcentaje de proyectos fracasan en su mayoría son por problemas de comunicación y de gestión. De lo que fácilmente podemos concluir que las metodologías no son perfectas y deben seguir un mejoramiento continuo.

El XP debe su producción de software a una fuerte arquitectura con lanzamientos de productos constantemente, para verificar y afinar la calidad del mismo. En vez de dar una fecha muy lejana en el futuro, el cliente puede ver resultados en un ciclo rápido de desarrollo, y de esta manera los desarrolladores pueden responder con confianza a las necesidades cambiantes del proyecto. Con esta idea, podemos considerar al XP como un rompecabezas que está conformado por pedazos pequeños, refiriéndose a los continuos lanzamientos, que por sí solos no tienen sentido pero que juntos y combinados como un todo forman un cuadro completo.

Uno de los aspectos más interesantes del XP es la simplicidad de sus reglas, que incluso pueden parecer básicas hasta ingenuas al principio, pero están basadas en sólidos valores y principios. Estas prácticas simples llevan al equipo de desarrolladores a ver donde están y así poder ajustarse a la situación particular del proyecto donde se encuentran. El XP al utilizar equipos pequeños genera también un ahorro de dinero en la parte de comunicación ya que los

desarrolladores solo necesitan relación directa con los integrantes de su grupo, con lo que también se logra una asociación compacta que siempre está en movimiento para realizar los objetivos los más rápido posible y ofrecer valor que el cliente quiere en su aplicación.

El desarrollo de software hoy en día es muy riesgoso, los clientes se encuentran insatisfechos, realizar una planificación a futuro cuesta mucho dinero en la cual se estima alcance y recursos, no se llega hasta donde se quiere y además se necesita mucho más dinero y personal de los que se establecía en un principio. Por ésto es que el modelado inicial se convierte en un proyecto distante y especulativo en el cual sin duda se equivocará, más aún en el ámbito empresarial y del estado en el que las reglas del negocio pueden cambiar repentinamente sin anticipación, refiriéndose a leyes, impuestos, etc. El XP puede detectar un fallo en el ciclo más cercano con lo que simboliza que cada error encontrado a tiempo es dinero ahorrado.

Las herramientas actuales que se usan para el desarrollo de software (tanto en la parte de planificación y el desarrollo en sí) han aumentando increíblemente, esto ha logrado reducir dramáticamente los tiempos de compilaciones y pruebas que consiguen afinar las proyecciones, el XP y las metodologías ágiles se adaptan y evolucionan constantemente, no así las metodologías tradicionales. El XP promueve el desarrollo incremental con el que el cliente al poco tiempo ya ve resultados, tal vez no lo esperado pero si algo en que basarse para el producto final. Cada pequeño éxito profundiza el respeto que se tienen por las contribuciones hechas por todos y cada uno de los miembros del equipo, con esta base los programadores son capaces de responder rápida y ágilmente a las necesidades de cualquier aspecto.

Es así como el XP ha logrado destacar en numerosos proyectos a nivel mundial y llevando el desarrollo de software de una manera sencilla y eficaz, dejando de lado todo lo innecesario que dificulta el trabajo. Hay diferentes comunidades que comparten sus experiencias para poder así enriquecer esta metodología que es considerada el núcleo de la evolución.

1.1.3 Actividades de XP

El XP realiza distintas actividades que se pueden agrupar en las siguientes:

- **Codificar.-** Es necesario plasmar nuestras ideas en forma de código, lo cual expresa la interpretación personal de la solución del problema que es en este caso el software, lo que se debe tener en cuenta es que al final del día se tiene que conseguir algo que funcione y represente un progreso para el producto. Al unir el código con otra persona podemos hacer comunes las ideas y de esta manera aprender y mejorar.
- **Realizar pruebas.-** Todo lo que se desarrolla debe ser probado su funcionamiento. Se deben desarrollar pruebas antes de escribir el código, es decir cuando se conoce los casos de uso (en XP se conocen como historias del usuario, que se explicará más adelante) para conseguir una prueba más objetiva de cómo está reaccionando lo realizado. Una vez que no se puede desarrollar otra prueba que haga fallar el código se habrá acabado esta actividad.
- **Escuchar.-** Es fundamental escuchar, ya que al desarrollar el software para el proyecto es lo que el programador entiende, esta idea se basa en lo escuchado al cliente, jefe del proyecto y todo el mundo en general. Se entiende que la primera vez no va a ser perfecto el lanzamiento pero se podrá verificar si lo obtenido es lo deseado. Según vaya habiendo nuevos ciclos donde se escuchará y delineará mejor los requerimientos se perfeccionará el producto.
- **Diseñar.-** El diseño es el modelo, donde se basa toda la lógica del sistema, teniendo un buen diseño va a ser que el proyecto se desarrolle con rapidez y eficacia. Se propone que el diseño sea lo más sencillo posible y en sectores que sean complejos se recomienda dividirlo en partes más pequeñas para conseguir su sencillez. El diseño también debe ser incremental para poder basarse en la metodología XP, debe cubrir las necesidades que el problema requiere, mas no lo que se cree que debería ser el diseño.

En una pequeña conclusión tenemos que, es necesario codificar ya que el código es el que hace el programa, también debemos realizar pruebas para saber que si lo que se ha avanzado

está bien codificado, debemos escuchar para saber que debemos codificar y probar. Por último el diseño debe representar lo codificado, probado y escuchado.

1.1.4 Valores del XP

Los valores que son tomados como base para poder realizar la metodología XP son:

Comunicación

XP lleva la comunicación al extremo, entre todas las partes que conforman el proyecto. El cliente debe formar parte del equipo para poder formular plenamente las necesidades del lado del negocio. En XP se puede trabajar en parejas y se dispone que se roten los puestos dentro del proyecto para que así el código sea comunal y se pueda modificar código de otros miembros. La comunicación promueve también el esparcimiento por el grupo del conocimiento técnico, para así cuando un problema aparezca todo el equipo pueda formar parte de la solución.

También las personas que tengan menor experiencia pueden aprender de los desarrolladores que tengan mayor práctica, lo que reduce el riesgo drásticamente en ciertas áreas.

Simplicidad

Ayuda al desarrollador a encontrar las soluciones más simples para lo que el cliente necesita, lo necesario es hecho y lo demás es desechado.

Se crean características en el diseño que solucionen los requerimientos actuales ya que el futuro no forma parte del proyecto, es decir, solo se ocupan de las necesidades inmediatas, que incrementalmente van a llegar al producto final, por que el futuro no tiene una fiabilidad total que se transforma en una pérdida de tiempo y a su vez de dinero.

Retroalimentación

El XP se esfuerza por realizar la retroalimentación tan rápido como se pueda, apenas se introduce el error hay que tratar de arreglarlo. Los cambios que se realizan cada cierto tiempo deben ser revisados para no tener efectos colaterales y enseguida deben ser unidos al resto de partes. Con pruebas de aceptación se consigue saber si el cliente obtuvo lo que deseaba, sino debe ser corregido.

Coraje

Se debe decir la verdad en todo momento, refiriéndose a las horas que hemos trabajado y el porcentaje que hemos avanzado, que a la final se van a ir ajustando para avanzar correctamente.

Se debe enfocar en que si un código no puede ser corregido para lograr su fin, este debe desecharse y volverse a desarrollar desde cero. Es difícil tomar este tipo de decisiones pero es lo necesario para llevar el proyecto al éxito. No se deben documentar excusas para tratar de opacar el fracaso, teniendo siempre la intención de alcanzar la eficiencia total. Se debe adaptar a todo cambio cada vez que se produzca.

1.1.5 Buenas prácticas del XP

El XP utiliza un conjunto de actividades que ayudan al desarrollo del proyecto, estas buenas prácticas son las siguientes:

- **El Juego de Planificación.-** Son una serie de actividades planificadas que son practicadas durante el transcurso del proyecto. El primer juego se lo realiza al inicio del proyecto y se incluye una sesión con el cliente y las personas relevantes (directivos y desarrolladores) para definir los requisitos iniciales y el contorno (conocidas como "historias de usuarios"). Cada historia de usuario se divide en tareas de unidades pequeñas de 1 a 5 días de trabajo, las tareas deben tener su prioridad con su respectivo desarrollo incremental. Si el juego sigue, se continúa con un "juego de lanzamiento" donde el cliente realiza un calendario de iteración de desarrollo que varían de entre 2 a 4 semanas, estas iteraciones tienen más historias que se basan en prioridad del negocio. En cada iteración el cliente puede cambiar historias, horarios y prioridades. El objetivo del juego es maximizar el valor del software que se está produciendo, lo principal es realizar las partes más importantes lo más rápido posible y ponerlas en producción.
- **Desarrollo de Probar-Conducir.-** Se manejan 2 tipos de pruebas, las de aceptación escrita por el cliente y la de unidad escritas por los desarrolladores. Las de aceptación

prueban las funcionalidades del sistema y las de unidad prueban que el código esté escrito según lo esperado.

- **Programación en parejas.-** El XP propone que todo el código desarrollado sea producido por una pareja que comparte un solo computador, con un solo monitor, mouse y teclado. Cada uno tienen una función diferente, uno realiza la función de conductor que escribe el código y la otra persona como su navegador que se encarga de observar que está sucediendo y realiza consejos estratégicos para desenvolverse de manera eficiente. El conductor por otro lado también debe explicar al navegador que es lo que está realizando, lo que logra que el código se revise mientras se produce; las dos personas pueden cambiar de funciones cuando sea necesario, también se interactúa con otros desarrolladores con el fin de realizar el requerimiento lo mejor posible.
- **Refactorización.-** El código se debe poder modificar sin que afecte su funcionalidad. Con ésto podrá ser mejorado cuando sea necesario, como por ejemplo se debe eliminar código duplicado, simplificar funciones, si un código se está volviendo complejo se lo debe cambiar en su diseño a uno más simple.
- **Diseño simple.-** Los desarrolladores deben seguir el principio de YAGNI – "No vas a necesitarlo" donde el diseño debe estar con la máxima simplicidad plasmado en tarjetas CRC (Clase-Responsabilidad-Colaboración). Solo se debe producir lo que se indica en las historias dadas, no se agregan características no necesarias, de esta forma las clases descubiertas durante el análisis pueden saberse si son necesarias o no para el producto final.
- **Propiedad colectiva del código.-** Se basa en que "Nadie es dueño del código", es decir que todos los desarrolladores son propietarios del código, pudiendo realizar cualquier cambio en cualquier momento aunque haya sido hecho por otra pareja de programadores, siempre y cuando se utilicen los estándares y los test de pruebas funcionen en su totalidad para tener una integración exitosa.
- **Integración continua.-** Todos los cambios se introducen en el sistema, por lo menos una vez al día. Si al integrar el código (que debió ser probado unitariamente) con el resto del sistema producido sigue funcionando correctamente esta parte se habrá

cumplido, de lo contrario se habrán encontrado errores en una etapa temprana que es lo que se propone. Se deben corregir estos errores y volver a hacer la integración si esto no funciona se borrará y comenzar de cero. En esta etapa debe estar presente el cliente para verificar la integración.

- **Cliente en el sitio.-** El equipo de desarrollo debe tener comunicación directa y continua con el cliente, este debe poner en claro las dudas que tenga sobre las historias, discutir el desarrollo de los lanzamientos, fijar prioridades, etc. En fin, el cliente debe proporcionar retroalimentación. A veces es difícil encontrar un cliente sénior ya que no está disponible, un junior no sirve, lo ideal es un cliente analista.
- **Lanzamientos pequeños.-** Un sistema rápido y eficiente se pone en producción en cada iteración, donde el cliente lo observa. El primer lanzamiento deberá tener un conjunto mínimo de requisitos pero vitales para el producto final. Todos los lanzamientos deben tener funcionalidades valiosas para el cliente, donde se consigue retroalimentación.
- **Semana de 40 horas.-** Los desarrolladores no deben ser héroes y acabar su trabajo en un día. Se debe plantear semanas solo de 40 horas de trabajo, si es necesario hacer horas extras esto solo se debe realizar dos semanas seguidas, ya que el producto comienza a decaer en calidad. El programador no debe estar obsesionado con el proyecto debe balancear su vida con el trabajo.
- **Estándares de codificación.-** Todo el proyecto debe ser entendible para el grupo de desarrollo, la única forma de lograr esto es con un estándar de desarrollo. El código debe estar escrito de manera clara y constante logrando una uniformidad tanto que no se distinga quien ha realizado un sección puntual del proyecto.
- **Metáfora del sistema.-** Todos los proyectos son guiados por una historia escrita en lenguaje común entendible para los desarrolladores y para el cliente, donde se explica cómo funciona el sistema en general, esto reemplaza a la arquitectura.
- **Reunión.-** Una reunión de 15 minutos diaria antes de comenzar a trabajar es necesaria para discutir los problemas encontrados en día anterior y definir lo que se va a realizar durante ese día.

- **Pruebas continuas.-** Las pruebas deben realizarse antes de escribir el código. Las pruebas deben ser escritas por los desarrolladores y también por el cliente.

1.1.6 Ciclo de vida del XP

El ciclo de vida del XP al igual que todas las metodologías se basa en entender lo que el cliente necesita, estimar los recursos y de esta forma crear la solución para entregar un producto final, pero XP propone un ciclo de vida más interactivo e incremental en su desarrollo. La mayoría de los proyectos se caracterizan por tener clientes que no siempre tienen definido desde el principio lo que quieren, por esto se trata de realizar ciclos de desarrollo cortos, llamados en XP "iteraciones".

Las iteraciones son un período de tiempo donde se desarrollan un conjunto de funcionalidades de software que están definidas por el usuario como "historias de usuarios". Cada iteración tiene análisis, diseño, desarrollo y pruebas pero con las prácticas que utiliza el XP.

Un proyecto tienen alrededor de 10 y 15 iteraciones, entre más rápido se la entreguen al cliente se va a tener más retroalimentación para obtener un producto que va representar firmemente lo que se pidió. Hay una fase inicial donde se programa cada iteración y se las superpone para que no se distancien en el tiempo.

Fase de exploración.- Es donde se establece el alcance general de todo el proyecto, aquí el cliente define lo que necesita mediante las historias del usuario para así poder realizar la primera entrega del producto. Los desarrolladores en esta fase comienzan con estimaciones de tiempo para la entrega (estas estimaciones deben afinarse en cada iteración), también comienzan a reconocer las herramientas, tecnologías y prácticas que se van a utilizar en el proyecto, es decir, se familiarizan con el entorno del proyecto. Para probar la arquitectura se realiza un prototipo y revisar si se ajusta lo que quiere el cliente, esta fase dura unas pocas semanas con un resultado de una visión del sistema con un plazo para obtenerlo.

Fase de planeamiento.- En esta fase se acuerda el tiempo que va a tomar realizar las historias del usuario, con un cronograma en el que participan el cliente, los gerentes y el grupo de

desarrollo que da como resultado el Plan de Entregas. El ciclo debe tomar solo un par de días, las pruebas para cada iteración son puestas en marcha al final de cada ciclo. Al final de la última iteración se debe tener listo el producto terminado.

Fase de iteración.- Es la fase donde se desarrolla en sí, al final de cada ciclo se debe tener un producto funcional que es parte del producto final. Este producto tiene las historias del usuario las cuales no tienen tanto detalle por lo que se requiere la participación del cliente de manera activa. También se puede medir el avance del proyecto con las iteraciones, una iteración sin errores es un avance significativo.

Fase de producción.- Aunque después de cada iteración se entrega un producto funcional al cliente, este puede pedir que el producto no se ponga en producción hasta no tener y comprobar la funcionalidad en su totalidad, además que lo cambios realizados puede determinarse si van a ser parte o no del producto final, en esta fase no se desarrolla nuevas funcionalidades solo hay tareas de ajuste. El primer producto de la primera iteración se debe mantener siempre en funcionamiento para ir agregando las nuevas fases hasta tener el software definitivo.

Fase de mantenimiento.- Esta etapa puede ser un poco dificultosa ya que hay que cumplir los cambios que requiera el cliente y el proyecto puede perder el ritmo. Se puede necesitar incluir nuevo personal para cambiar la estructura del equipo.

Fase de muerte.- Ocurre cuando el cliente ya no tiene más historias de usuarios que se necesiten incluir en el sistemas, para esto los desarrolladores deben enfocarse en otros aspectos como lo son el rendimiento y la confiabilidad del sistema. Aquí se dan los pasos finales que son la documentación final del proyecto, no se hace cambios en la estructura del sistema. La fase de muerte también puede pasar cuando no hay beneficios buscados por el cliente y se detiene el proyecto así como también cuando no hay presupuesto para seguir.

1.1.7 Actores y responsabilidades del XP

Cada proyecto XP tiene varios roles diferentes, cada uno con sus propias reglas y responsabilidades únicas. XP intenta mejorar la comunicación para que exista relación directa entre clientes y desarrolladores. Los roles logran que los objetivos sean comunes así que si se quiere terminar un ciclo o fase se debe hacer en grupo. XP ofrece a los desarrolladores la autoridad para tomar decisiones técnicas y a los clientes a tomar decisiones de negocios, definiendo los roles y actividades de los mismos se desarrollará mejor el proyecto.

- **El cliente.-** Es el impulsador del proyecto, este define el proyecto y establece sus objetivos. Entre más frecuente sea su participación en el proyecto más posibilidades hay que se finaliza con éxito. El cliente toma las decisiones del negocio ya que sus derechos y obligaciones son sobre su conocimiento sobre el negocio. El cliente también representa al usuario final ya que a través de él, es que se sabe las características que ellos realmente necesitan, desde su perspectiva. Escribe la "historias de usuarios" y decide en que iteración se deben implementar dependiendo en que valor pueden aportar al producto final, además realiza las pruebas funcionales para comprobar lo que se ha desarrollado. El cliente también representa a los intereses del negocio, que pagan por el proyecto, tratando de maximizar sus inversiones. El cliente tiene una posición de autoridad con la potestad de decidir qué se debe hacer.

- **Programador.-** Básicamente el trabajo del programador es convertir las historias de usuarios en un código funcional, de la compresión del problema del programador es que se va a dar la solución. A partir de las historias los desarrolladores deben estimar un tiempo estimado, en base a experiencia del proyecto hasta el momento. Las estimaciones en un principio no son muy acertadas pero se van afinando durante el proyecto, además ayudan al cliente a programar las siguientes iteraciones. Los desarrolladores deben crear y mantener el sistema durante cada iteración es decir mientras evoluciona, realizando solamente lo necesario para mantener el proyecto lo

más simple y valioso posible. El programar además de comunicación y coordinación con el cliente debe tenerla con el resto del equipo, en el XP no se distingue entre analista, diseñadores y codificadores ya que todos realizan todas las funciones.

- **Entrenador.-** Algunos proyectos de XP tienen un entrenador, que se definen como mentores y guías del equipo. Su posición es de predicar con el ejemplo, aunque a veces es difícil ponerlo en práctica de una forma versátil, ya que se requiere la sabiduría de un maestro que conozca a fondo el proceso de XP. La principal virtud es la de la experiencia. La enseñanza que se imparte a veces es de forma directa y sino durante el desarrollo, además se puede sugerir nuevas prácticas, ofrecer ideas para resolver problemas técnicos difíciles o servir de intermediario entre el equipo y la administración. El entrenador tiende a estar en segundo plano mientras el equipo evoluciona.

- **Rastreador.-** El rastreador mantiene un seguimiento a la programación, su responsabilidad está en definir en qué grado de aceptación se encuentra las estimaciones realizadas contra el tiempo real que se ha utilizado, del mismo modo recopila información de cambios de velocidad, de las horas extras trabajadas y sobre las pruebas realizadas, sus resultados, etc. También realiza un seguimiento al progreso de cada iteración para comprobar si se ajusta al calendario establecido, donde se indican los cambios de comportamiento que puede afectar a la programación, toda esta información es utilizada para mejorar en futuras estimaciones. La forma de realizar su trabajo es cada uno o dos días le pide a cada desarrollador cuantas horas ha completado, lo realiza de manera informal y cómodamente para conseguir la honestidad en la respuesta.

- **Probador.**- Aunque los desarrolladores también realizan esta función, a veces se designa una persona específica para realizar las pruebas y además ayudar al cliente a realizar las pruebas funcionales. También difunde los resultados al equipo para seguir adelante si todo está bien y de lo contrario rectificar los errores.

- **Consultor.-** Se trata de una persona no que está directamente en el equipo, un miembro externo, ayuda con un conocimiento específico en algún tema que sea necesario para el proyecto.

1.1.8 Herramientas de análisis y diseño del XP

El XP es considerado como una metodología ágil para el desarrollo de software, pero también debe tener ciertas tareas que hay en las metodólogas tradicionales como el análisis y el diseño, la diferencia radica en que se lo realiza de una manera diferente con un enfoque más ligero y transparente. Las herramientas para realizar estas tareas son:

Historias de Usuarios

Es la técnica que utiliza el XP para definir los requerimientos que va a tener el software a desarrollarse. Son tarjetas de papel que son escritas por los cliente donde se describe brevemente las características que el sistema debe tener, ya sean estos funcionales o no funcionales. La descripción utiliza terminología del cliente sin lenguaje técnico, deben proporcionar suficiente detalle para poder definir una estimación de cuánto tiempo se requiere cada historia, las estimaciones son de 10 horas a dos semanas, si son mayores a estas quiere decir que son muy complejas y deben dividirse en historias más pequeñas. También sirven para desarrollar un plan de entregas, reemplazan a un complicado y extenso documento de requisitos.

Las historias de usuario se deben manejar de manera dinámica y flexible, es decir, en cualquier momento se puede destruir, cambiar, reemplazar por unas más específicas o también generales, añadirse más o ser modificadas.

Respecto a las plantillas de las historias de usuario no existe consenso, pero en muchos casos solo se proponen 3 aspectos básicos:

- *Tarjeta.-* Donde se describe la información para identificar y detallar la historia
- *Conversación.-* Es la que mantienen los programadores y clientes para ampliar y detallar la información

- *Pruebas de aceptación.*- Es la confirmación de que la historia se ha implementado de manera correcta y de acuerdo al cliente.

Una limitación es que al ser escrita en lenguaje común, se prestan a muchas interpretaciones, lo que puede producir contratiempos, por lo que se necesita desarrolladores competentes.

Modelo de Negocio

Es muy importante en un proyecto que el software creado esté alineado correctamente con el modelo del negocio donde se va a implementar.

Es muy difícil conseguir ajustarse, ya que los modelos de negocios cambian y las empresas deben actuar cada vez más rápido y contra más elementos, como la tecnología nueva, competitividad, leyes, etc. Por esta razón el sistema tiene que ser capaz de responder de manera eficiente contra estos cambios y satisfacer las necesidades del cliente.

Plan de entregas

Se establece a partir de las historias del usuario que serán agrupadas y organizadas para conformar una iteración. Para programar la entrega se deben reunir a todas las personas que participan en el proyecto, es decir, el cliente, desarrolladores y administrativos.

El cronograma se realizara a partir de estimaciones de tiempo de desarrollo por parte de los programadores, luego de algunas iteraciones es recomendable revisar el plan de entregas para ver sus resultados y ajustarlos si es necesario.

UML

La realización de un proyecto con la metodología XP no tiene relación con la utilización de UML para análisis o el diseño. Pero debido a que los programadores que utilizan XP son en su mayoría los que utilizan programación orientada a objetos, es muy común observar diagramas UML con proyectos XP. Aunque esto no quiere decir que sea necesario, resulta muy útil dependiendo del tipo de proyecto que se está desarrollando; de lo que podemos concluir que XP a pesar de tener una serie de principios básicos que no puede ser ignorados también se ajusta a utilizar otros principios que nos ayudan en nuestro trabajo y no pueden ser desestimadas.

1.1.9 Aplicabilidad del XP

Como es de razonamiento lógico, una sola metodología no va a servir para todos los proyectos de desarrollo de software, cada uno tiene ciertos escenarios de aplicabilidad. En proyectos denominados grandes, con extensos grupos de trabajos y donde el cliente sabe exactamente lo que quiere desde un principio, las metodologías tradicionales son la mejor opción. En cambio donde son proyectos medianos y pequeños donde se tienen pocos programadores y el cliente no sabe lo que realmente quiere hasta cuando el proyecto está en marcha, es recomendada una metodología ágil como el XP.

Sobre el equipo de trabajo de XP no se define un número exacto de personas, pero en la práctica todo coincide en que son menos de 20. Es entendible ya que por ejemplo las reuniones diarias con más de 20 personas no es algo concebible.

El XP puede ser utilizados con otras prácticas, así también se puede trabajar con sistemas de calidad como el ISO 9000 y el CMMI.

En el aspecto físico también se debe prestar atención ya que el lugar de trabajo debe estar acorde a la metodología, por ejemplo de debe tener un ordenador con dos sillas (programación en parejas), mesas redondas para las reuniones de equipo, aparatos que permitan la constante comunicación entre los miembros del grupo, son algunos de los requerimientos que se necesita para seguir adelante con la metodología.

El cliente debe entender la metodología para así poder seguirla y participar activamente de ella, además de destinar recursos necesarios. Por último todos los participantes deben estar compenetrados con el proyecto y la metodología, deben compartir lo que han realizado y ser responsable por los cambios que hicieron a otras partes del sistema.

1.1.10 Críticas al XP

Las metodologías no son perfectas también tienen sus críticas, XP por su parte tiene algunos aspectos que son desacreditados:

- La programación de parejas sería un poco disfuncional debido a que ciertos programadores tienen sentimientos posesivos sobre su código, lo que dificulta que otros desarrolladores lo puedan modificar para su mejoramiento.

- La estimación de costos es otro problema, ya que XP se ajusta a todo tipo de cambios, lo que cambiarían constantemente los costos, lo que deja como saldo estimaciones fracasadas. Para software que son para la misma empresa que los desarrolla, esto no es un gran inconveniente, pero si para empresas que se dedican a desarrollar para otras empresas donde se debe presupuestar y sacar ganancia de sus proyectos.
- En un mundo tan competitivo, donde se necesita todo listo instantáneamente es un lujo trabajar solo las 40 horas semanales, más aún cuando hay grandes cambios en el transcurso del desarrollo.
- Una crítica también es sobre los diseños que casi siempre solo se definen para el inmediato siguiente, y no a largo plazo. XP como ya lo hemos dichos se ajusta a los cambios lo que hace que sea casi imposible preveer lo que pasará, esto puede ser que demore el proyecto en cada iteración, sin tener un camino definido.

También podemos decir que todas las prácticas definidas por XP no son una invención del método, sino una recopilación de todas, y que cada proyecto está en un escenario diferente y se debe saber si esta metodología se puede utilizar para ese desarrollo, ya que tiene lanzamientos de pequeños proyectos en plazos cortos para adaptarse a los cambios de requerimientos que se pueden dar por los clientes, usuarios, etc.

1.2 SOFTWARE LIBRE

1.2.1 Introducción del Software Libre

El software es un conjunto de sentencias (instrucciones) que un computador puede interpretar y ejecutar para realizar tareas específicas, la libertad es la facultad que permite decidir entre realizar una determinada acción o no llevarla a cabo. Lo que nos lleva al software libre es aquel que no esconde su código, como su nombre lo dice lo deja libre, permite realizar con él lo que se desee. En la actualidad con una generalización de la informática en casi todas las actividades que realizan las personas, se tienen limitaciones para las soluciones que se han implementado como patentes y restricciones en el software.

Sobre su nombre tiene cierta ambigüedad, ya que en inglés sería "free software", y la palabra *free* significa tanto gratis, como libre. Pero en español no existe tal problema, y se sabe que es software libre y no software gratis. Aunque el software libre suele estar disponible gratuitamente o por lo menos al precio de su distribución, no es obligatorio que sea así.

El software gratis es conocido como *freeware*, el software libre que es comercializado es conocido como "software comercial", éste debe conservar sus derechos de modificación y redistribución. Tampoco debe confundirse con el "software de dominio público" que es aquel que no requiere licencia, ya que sus derechos de explotación pertenecen a la humanidad, y pertenece a cada uno por igual. Lo contario a software libre es el software licenciado donde se priva de las libertades y se restringe todo mediante licencias que indican todas sus limitaciones.

1.2.2 Historia del Software Libre

A pesar de que la historia de la informática se podría considerar corta, no es el caso del software libre ya que es una de las más largas del campo. Se podría decir que el software nació libre y luego en la juventud fue privado para en su madurez tratar de volver a la libertad, como se va a explicar.

Las primeras comunidades de software, aunque no reconocidas de esa forma, ya existían desde hace mucho tiempo. Entre los años 1950 y 1960 casi todo lo que se desarrollaba en software era realizado por un grupo de académicos que no lo veían como un producto, ya que los programas eran distribuidos y mantenidos por las comunidades sin ningún tipo de restricción. El código fuente iba con el software final, para que los usuarios lo modificaran con los errores que iban encontrando, entiéndase que en esa época los usuarios también eran expertos en el tema.

A finales de 1960 comenzó el cambio donde los sistemas operativos y los compiladores de los lenguajes de programación evolucionaron, las empresas comenzaron a invertir en software, lo que hizo que sus costos se elevaran. El software estaba compitiendo con el software incluido con el hardware de los fabricantes, es decir que se vendía las máquinas con el software precargado, no se incluía el costo del soporte lo que equivalía más gastos. Los usuarios que podían ellos mismos satisfacer sus necesidades de mejor forma no querían que se vendiera el hardware con el software incluido, por esta razón Estados Unidos demandó a IBM ya que era anticompetitivo para el resto del mercado. En 1970 IBM comenzó a distribuir las primeras versiones de UNIX, que es un sistema operativo, sin costo al gobierno y a investigadores académicos, no se podía considerar software libre ya que no se podía distribuir, y tampoco distribuir versiones modificadas.

A finales de los años 1970 y el principio de los años 1980, las compañías de software ya cobraban por las licencias de software de manera común, llamándolos "Productos Informáticos", con todas las restricciones que tiene en la actualidad el software licenciado. En esta época Richard Stallman trabajaba en el laboratorio del Instituto Tecnológico de Massachussets (MIT, por sus siglas en inglés) en donde habían recibido una impresora como donación de una empresa externa. La impresora era utilizada por todos los empleados para imprimir, pero parecía tener ciertos problemas, se atascaba el papel seguidamente; el problema radicaba en que no se emitía ninguna señal o alerta indicando este problema a los usuarios lo que no dejaba culminar la impresión. Entonces los empleados al ir a buscar sus trabajos, recién se enteraban del problema, daba como resultado una gran pérdida de tiempo, el que aumentaba cuando se acumulaba el trabajo debido a los estancamientos. Es por esto que entra Stallman a

arreglar el inconveniente enviando por red una notificación de atascamiento, para este fin necesitaba el código fuente de los controladores de la impresora, la empresa se negó a entregárselo, aunque lo que iba a realizar era sin ningún fin lucrativo. Justo aquí es cuando él decide a no dejarse imponer restricciones innecesarias y que no dejaban continuar con el progreso y deja sentado un precedente para que en 1984 comience a trabajar en el proyecto GNU, que tenía como idea ser un sistema operativo completamente libre, y un año más tarde en 1985 se fundó la Free Software Foundation (FSF), donde Stallman ya nombraba al software libre y la definición de "copyleft" que daban la libertades a los usuarios y también no permitía que se apropien del software.

Se debe también mencionar un hecho sucedido en la década de 1970, cuando todavía se daba en el MIT el hábito de repartir el código fuente de los programas entre los desarrolladores y profesores a través del campus. Es así como Bill Gates y Paul Allen que luego serían los fundadores de Microsoft, escriben un código para un sistema operativo de una máquina Altarir 8800, como era costumbre alguien copia el código y lo reparte en el campus de la universidad, Gates sumido en la cólera escribe la famosa "Carta abierta a los aficionados", donde menciona que los estudiantes se roban los programas entre sí, lo que hace que no puedan escribir bueno programas y que permite realizar un trabajo profesional por nada. Se considera que desde aquí quedan sentadas las diferencias de Gates con el software libre.

El surgimiento del internet fue uno de los detonantes del esparcimiento del software libre ya que permitía su distribución rápida y además colaboración entre los desarrolladores. Se podían hacer programas entre personas que no se encontraban en la misma oficina, además de las herramientas que brindaba como correo electrónico o servidores de archivos.

1.2.3 Definición del Software Libre

Su definición se refiere a un software que tiene todas las libertades de realizar adaptación y distribución, garantizando a los usuarios la posibilidad de usarlo para lo que se requiera y necesiten, dando a conocer como funciones y adaptarlo a las necesidades. También se puede compartir legalmente las copias con las personas que se desee, ya sea de la manera que se recibió o con modificaciones.

El software libre se basa en la ideología que el software es conocimiento y difundir sin trabas, su ocultación es una actitud antisocial y la posibilidad de modificar programas es una forma de libertad de expresión.

El software libre trae consigo grandes ventajas como lo son la colaboración para el desarrollo, la fácil repartición de los programas, entre otras. Por otro lado tiene pocas desventajas que en su mayoría son sobredimensionadas por la competencia, aunque la financiera es una de las fuertes, ya que la distribución la puede realizar cualquier persona y casi siempre es alguien distinto al autor, para poder obtener dinero se debe tener un modelo de negocio y mecanismos de financiación. En el área de soporte es donde se puede beneficiar económicamente ya que no hay muchas personas especializadas en el tema.

En el sector público es un gran beneficiario del software libre ya que primero se ahorra una gran cantidad en dinero respecto a tener software licenciado, y además puede ajustar los programas a sus necesidades de necesitar permisos.

1.2.4 Cuatro libertades del Software Libre

De manera general se podría decir que un software se considera libre si el autor entrega el código fuente sin pedir ningún acuerdo legal de restricciones sobre lo que se haga con éste, de una manera clara para esto se tiene cuatro libertades las cuales deben cumplirse para considerarse software libre.

1. *La libertad de usar el programa, con cualquier propósito*.- Es decir que se pueda utilizar el programa con cualquier fin, ya sea este educativo, comercial, político, cultural, de entretenimiento, etc. Esta libertad se da para contrarrestar cierto software que tiene licencias que solo dejan utilizarlo de manera específica y asociado a ciertas áreas.
2. *La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades*.- Para esta libertad se necesita el acceso al código fuente ya que de esta forma es cómo se va a poder describir funciones ocultas, saber cómo es que se realizan las funciones y métodos del sistema para mejorarlo o quitar partes que no sean necesarias para la persona que lo utilice.

3. *La libertad de distribuir copias del programa, con el cual puedes ayudar a tu prójimo.*- Se es libre de redistribuir el programa aunque uno no sea el dueño, de manera gratuita o con algún costo, en cualquier medio ya sea por mail, por FTP, puede ser al número de personas que se quiera.
4. *La libertad de mejorar el programa y hacer públicos las mejoras a los demás, de modo que la comunidad se beneficie.*- Por ejemplo se puede mejorar el tiempo de respuesta del programa, hacerlo más liviano para que use menos recursos de hardware o que no tenga errores, etc. Y así mismo también puede repartirlo por internet en foros, o en sitios web.

Las libertades se complementan unas a otras, es por esto que se necesitan de las cuatro para que un programa sea declarado como software libre. Por ejemplo para las libertades 2 y 4 se necesita tener el código fuente para poder adaptarlo y mejorarlo, las libertades 1, 3 y 4 hablan de la posibilidad de distribuir los programas y con cualquier fin sin que ninguna licencia lo restrinja.

Para que estas libertades sean reales deben ser irrevocables, mientras no se tenga mal o incorrecto uso, si el desarrollador tiene el poder de revocar las libertades esto deja de ser software libre. Los derechos de autor no permiten la modificación, duplicación y redistribución lo que va en contra de las cuatro libertades, lo que nos dice que el software libre objeta la mayoría de estos derechos reservados.

1.2.5 Licencias del Software Libre

Las licencias de software licenciado tratan sobre puntos similares al software libre , como la redistribución y las condiciones de uso, solo que los manejan de diferente forma.

La licencia tiene las normas de uso, a las que tienen que regirse las personas que lo van a utilizar, los distribuidores, los integradores y todo el que se relacione con el producto, en este caso el software. Estas normas o condiciones que se imponen son precisadas por los propios autores, que según la normativa de propiedad intelectual son los propietarios de la obra, en software libre no se considera esto ya que con un pequeño cambio en el programa ya se considera una nueva obra y se puede imponer otro tipo de licencia con términos diferentes.

Las licencias que se manejan en software libre son las siguientes:

Licencia GPL: La Licencia Publica General de GNU (GNU GPL por sus siglas en inglés), es la más conocida y utilizada de las licencias de software libre. Fue hecha por la Free Software Foundation (FSF) para ser utilizada en todo el software generado por la fundación. Esta licencia dice que el autor conserva los derechos de autor y permite la redistribución y modificación bajo los términos de la misma licencia o una compatible, lo que hace imposible crear un producto con partes que no pertenezcan a la licencia GPL. La licencia está desarrollada para asegurar que el software siempre sea libre, ya que un programa bajo esta licencia nunca podrá ser propietario porque es un tipo incompatible con la licencia GNU GPL, los detractores de esta licencia dicen el hecho de que solo se pueda asociar a sí misma o las compartibles, es un recorte de libertad, pero los partidarios dicen que es una manera de asegurarse de que siempre sea libre.

Licencia LGPL: La Licencia Publica Menor del proyecto GNU, esta es otra licencia del Free Software Foundation (FSF), al inicio fue creada para el software libre de las bibliotecas (la "L" es de la palabra en inglés *library*) permite la integración con cualquier clase de software así sea propietario. Su creación se dio enfocada en los programas de bibliotecas, para fomentar su uso y desarrollo sin tener problemas con la integración de la licencia GPL, pero su uso se dio para todos los tipos de programas que buscaban tener la posibilidad de integración. Es por esto que la FSF le cambió el significado a la hermana menor de la GPL (la "L" de *lesser*), indicando su utilización solo para casos puntuales y especiales. Se nombran algunos casos de software que no es para biblioteca y usa esta licencia como Mozzila.

Licencia BSD: La licencia Berkeley Software Distribution, es utilizada para todo el software realizado por la universidad Berkeley de California, Estados Unidos. Esta licencia solo indica que debe reconocerse el crédito de los autores originales, permite la distribución del código fuente, así como la aplicación final y se puede realizar modificaciones e integrar con otros programas. Su creación se dio para utilizarlo en el software creado por las universidades, en proyectos que eran financiados por el gobierno de EEUU. Ya que el proyecto había sido pagado antes de ser producido, con lo que no era necesario ningún cobro y cualquier empresa o persona pudiera hacer uso de este sin restricciones.

CDDL(Common Development and Distribution License): Una licencia creada por SUN en diciembre del 2004, cumple con todas las bases de software libre como utilizar, estudiar, modificar y distribuir ; pero a la vez agrega nuevas condiciones no incluidas en la mayoría de las licencias de código abierto, como lo es permitir cualquier tipo de código sin importar de qué tipo de licencia provenga. También la CDDL solo se aplica a la totalidad de una aplicación, y no así a las partes del mismo.

Licencia Creative Commons: Fue creada en el 2001 por expertos en propiedad intelectual, derecho en la sociedad de la información e informática. Se puede decir que son licencias "a la carta" ya que permiten a los autores seleccionar cual es tipo de libertades que desea para su obra, ajustándose a cuatro mandatos: reconocer al autor original, se permiten modificaciones al trabajo, se puede redistribuir el trabajo y además se puede cambiar el tipo de licencia. Esta licencia tiene el aval profesionales del derecho y que no solo es para software sino también para trabajos artísticos, literarios, didácticos, etc.

Podemos concluir que todas estas licencias de copyleft son una práctica donde el titular de los derechos de autor permite que su obra sea libre de distribución de copias y de otras versiones con modificaciones pero siempre y cuando estas tengan las mismas libertades.

1.2.6 Software Libre y el Código Abierto

Para poder entender la diferencia, primero se debe definir lo que es *Open Source* o software de código abierto.

El código abierto es una iniciativa de un grupo de personas interesadas en el software libre que quisieron darle a sus programas un término de mercadotecnia más amigable para las principales casas de software y demás empresas tecnológica, pero con menos carga ideológica contra el software propietario. Es de aquí que se crea el Open Source Initiative (OSI) por Bruce Prens y Erick S. Raymos y se distancia del software libre de Richard Stallman.

El código abierto se enfoca su prioridad en la excelencia técnica del software, tomando al código fuente como un medio para llegar a dicho fin; es decir que si se expone el código

fuente de los programas a cientos de usuarios, ellos podrán ayudar a corregir errores y hacer mejor tomándolo como beneficio propio, que a su vez ayuda a toda la comunidad de código abierto. Sobre la venta de los programas, es permitida siempre y cuando las modificaciones sean compartidas con la comunidad.

La OSI critica al software libre por no solucionar los problemas de ambigüedad que crea en inglés el *Free Software* ya que puede ser interpretado como software libre o software gratis, lo que quita competitividad al tratar de venderlo en el mercado.

La principal ventaja que obtienen las grandes empresas de que su código pueda ser manipulado, es la colaboración de la comunidad ya que 1000 o 5000 programadores tratando de mejor un programa es mejor que 50 o 100 especialista desarrolladores.

1.2.7 Java

Java es un lenguaje de programación orientado a objetos que fue desarrollado por Sun, como parte de un proyecto de investigación para desarrollar software para dispositivos de red y sistemas embebidos para competir con Microsoft, aunque antes de ésto se remonta a un programa de fácil interfaz para electrodomésticos. Para el proyecto se iba a utilizar C++ pero cuando se estaba desarrollando se crearon muchas dificultades que fueron acumulando, hasta que se tuvo que decidir crear una nueva plataforma desde cero. Para su creación se basaron en Eiffel, Smalltalk, Objetive C, Cedar.

1.2.7.1 Historia de Java

Java aparece en 1991 para ser parte de un proyecto con el fin de tener una interfaz más amigable y cómoda en los electrodomésticos como lavadoras, hornos microondas y principalmente televisores interactivos. Esta filial se llamó FirstPerson donde se intentaba desarrollar esta tecnología en un principio con C++ pero como se encontraron con una serie de dificultades, trataron de modificarlo y extenderlo, donde nació el C++ ++- (la parte de ++- era porque se colocaba nuevas características y se quitaban otras), esta idea se abandonó porque igual no cumplía con las expectativas y ahí es donde se comienza un nuevo lenguaje desde cero. El nuevo lenguaje tomo su nombre por un roble que se veía desde la oficina donde se realizaba el proyecto, de ahí su nombre Oak (roble en inglés). Sun al darse cuenta que este

proyecto no iba a ser un gran éxito y no iba a dar los beneficios comerciales que buscaban, presionaron a FirstPerson a desarrollar nuevas estrategias para poder obtener ganancias, lo que no fue conseguido y FirstPerson cerró en 1994.

Sun se encontró con este lenguaje, que podía ser utilizado para su progreso, primero le cambio el nombre por Java, debido a que ya existía un lenguaje con el nombre Oak. Sobre el nombre Java existen algunas teorías, una es que los miembros del equipo frecuentaban una cafetería donde se servía un tipo de café llamado Java , esta es la más fuerte ya que su icono es una taza de café; también se especula que es un acrónimo de sus creadores: James Gosling, Arthut Van Hoff, y Andy Bechtolsheim. La última teoría es que su nombre fue sacado de una lista aleatoria de palabras. Luego del cambio de nombre hizo varias modificaciones y vieron en el inminente despegue de internet una oportunidad para utilizar su lenguaje adaptándolo a navegadores y es así que se presentó en sociedad en mayo de 1995.

1.2.7.1 Definición de Java

Es un lenguaje de programación portable a toda plataforma, orientado a objetos y que permite desarrollar aplicaciones distribuidas. Sun describe a Java como "simple, distribuido, interpretado, robusto, seguro, de arquitectura neutral, portable, de altas prestaciones, multitarea y dinámico". Esta definición viene de:

- *Sencillo* para los desarrolladores no se requiere un gran entrenamiento para comenzarlo a utilizar, lo que son familiarizados con C++ puede migrar rápidamente a Java y ser productivos en poco tiempo
- *Distribuido* por que tiene una colección de clases para utilizar en aplicaciones en red, estos permiten abrir conexiones con servidores o cliente remotos y así tener las aplicaciones distribuidas.
- *Interpretado* ya que cualquier computadora que tenga su intérprete puede utilizar las aplicaciones creadas en Java
- *Robusto* debido a que simplifica la gestión de memoria, que libran a los programadores de una familia entera de errores como lo son la aritmética de los punteros de C.

- *Seguro* ya que implementa barreras de seguridad en el lenguaje y en el sistema de tiempo real. Lo consigue con fuertes restricciones al acceso a memoria, asegura que no se viole ninguna construcción del lenguaje, verificación de los nombres de las clases y sistema de seguridad de la interfaz que refuerza las medidas de seguridad en muchos niveles.
- *Arquitectura neutral* Java está diseñado para que sus aplicaciones puedan soportar redes hibridas, con diferente hardware y sistemas operativos.
- *Portable* debido a que reduce su código a un formato independiente de plataforma diseñado para transportar código de manera eficiente a través de múltiples plataformas.
- *Alto rendimiento* Java en sus últimas versiones tienen diversas herramientas de optimización
- *Multitarea* las aplicaciones que solo pueden ejecutar una sola aplicación a la vez se ven totalmente limitadas, por esto Java soporta sincronizaciones múltiples de hilos de ejecución a nivel del lenguaje para lograr actividades simultaneas en un programa. A través de procesos ligeros, que son piezas independientes de un gran proceso es como se puede realizar más de un proceso a la vez.
- *Dinámico* ya que no intenta utilizar todas las partes de las aplicaciones al mismo tiempo, sino cuando estas son llamadas o requeridas.

1.2.7.2 Característica de Java

Java tiene cinco objetivos principales por lo que fue creado:

- Se debe utilizar con la programación orientada a objetos
- Se debe poder ejecutar las aplicaciones en múltiples sistemas operativos
- Debe tener por defecto la capacidad de trabajo en red
- Se debe diseñar para ejecutar código en sistemas remotos de forma remoto
- Se utilizó lo mejor de otros lenguajes orientados a objetos.

Para entender mejor los objetivos principales se deben definir ciertos conceptos que son fundamentales:

1.2.7.2.1 Programación Orientada a Objetos

La Programación Orientada a Objetos (POO) nace de la necesidad de evitar que programadores construyan aplicaciones muy parecidas una y otra vez para los mismos problemas. Con POO que es una serie de normas para poder realizar las cosas de una manera de que todos los desarrolladores puedan utilizarlas y avanzar su trabajo, se logra que el código se pueda reutilizar. POO es un paradigma de la programación que utiliza una abstracción de la realidad (a los que llama objetos) para poder solucionar problemas, es decir, que se basa en la realidad y no en los términos del computador para poder realizar una aplicación.

1.2.7.2.1.1 Elementos de POO

POO al tratarse de una nueva forma de programar para encontrar soluciones a los problemas, introduce nuevos conceptos que debe entenderse para saber la manera en que se trabaja.

Objetos.- Es una abstracción de la realidad, es decir, en el mundo real se puede decir que objeto es cualquier cosa que puede ser observado, estudiado y aprendido. En POO se toma a los objetos como la representación de la vida real, con sus propiedades o atributos, su comportamiento o funcionalidad, los que reaccionan a los eventos. Los objetos son la base de los diferentes componentes, que tiene un rol especifico en el programa, que se comunican entren ellos con formas definidas con anterioridad.

Clases.- Las clases se pueden tratar como los moldes o las formas de los objetos, son una plantilla que se defina las variables y los métodos en común que comparte cierto tipo de objetos. Otras definiciones también establecen a las clases como las agrupaciones de los diferentes tipos objetos que se pueden abstraer de la realidad.

Método.- Es el comportamiento que va asociado a un objeto, es decir, son todas las acciones que realizar por sí mismos. Su utilización puede cambiar las propiedades del objeto, o generar un evento; los métodos son activados a través de un mensaje.

Atributos.- Los atributos son las características que describen al objeto, los cuales pueden ser visibles desde afuera del objeto, además de poder ser cambiado con un método. Los atributos tienen dos partes un nombre del atributo, y un valor del mismo.

Evento.- Los eventos son el medio con el cual las clases interactúan entre sí, o con el usuario, cambian los estados o avisan de su cambio y deciden como manejarlo.

Mensaje.- Es una solicitud para que un objeto realice un método. Esta solicitud debe contener el nombre del objeto, el nombre de la operación, un grupo de parámetros y a veces pide que se produzca una respuesta.

Identidad.- Los objetos tienen una identidad, la cual los distingue de los otros objetos. Casi siempre esta identidad es un código o número que lo identifica, y es independiente del estado. Casi nunca cambia durante la utilización del objeto.

1.2.7.2.1.2 Características de POO

La programación orientada a objetos tiene un serie de características que facilitan su fácil puesta en desarrollo, algunas ya eran utilizadas en otras formas de programación y otras son introducidas por este paradigma.

Abstracción.- Es la capacidad de los objetos de capturar los comportamientos, representando sus características principales sin utilizar las no esenciales. En este proceso se identifican los procesos, las funciones y los métodos para los cuales se utilizan una variedad de técnicas relevantes. La abstracción es una parte primordial de POO ya que a través de esta es que se arman las clases que van a ser utilizadas para la solución del problema.

Encapsulamiento.- Se puede explicar esto como una envoltura que tienen los objetos, que lo convierte en algo independiente, la envoltura oculta los detalles relacionados con la construcción del objeto, es decir, podemos obtener todos los métodos del objeto sin la necesidad de saber cuáles son los recursos que se utilizan para realizarlo. Con la encapsulación se garantiza la integridad de los datos, solo pudiendo acceder a ellos por los medios definidos (propiedades, métodos, eventos). Se tiende a confundir la encapsulación con el principio de ocultación ya que se utilizan en conjunto.

Herencia.- En POO todas las clases se relación entre sí, dependen de una jerarquía de clasificación. Los objetos se basan en una clase de base, superior en la jerarquía, heredan todas

sus propiedades, métodos y eventos. Una vez heredados se puede implementar o modificar. La herencia supone una característica para que objetos puedan compartir y extender su comportamiento sin que tengan que volver a ser implementados, lo que simplifica el código y ahorra recursos.

Polimorfismo.- Así como la herencia tiene objetos que comparte sus comportamientos, puede haber objetos con el mismo nombre pero con algunas características diferentes. El polimorfismo es la propiedad que indica que una entidad pueda tomar diferentes formas, la forma en cómo se llame al objeto es lo que indicará la forma de cómo se va a comportar.

Modularidad.- Es la capacidad que tiene de dividir las aplicaciones en partes más pequeñas, que son llamadas módulos, haciendo lo más independiente entre sí y de la aplicación. Estos módulos tienen relaciones para comunicarse, permiten la reutilización, facilitan la verificación, pruebas, depuración y manteniendo de cada uno.

Principio de ocultación.- Con esta característica se logra que los objetos no pueden ser modificados directamente y se obliga a que se utilice las funciones definidas para modificarlos (conocidas como interfaz). Se asegura la integridad de los datos ya que solo se puede acceder con lo planificado y no de manera inesperada cambiando su estado interno.

1.2.7.2.2 Independencia de Plataforma

La característica por la cual Java es reconocido y por la que puedo destacar entre los demás lenguajes de programación es que es independiente de plataforma. Lo que quiere decir que si se desarrolla un programa una vez, podrá funcionar en cualquier computador sin necesidad de hacer adaptaciones o modificaciones. Es por esto que uno de los lemas de Java dice *"write once, run anywhere"* ("escríbelo una vez, ejecútalo en cualquier parte").

Esta ventaja significa para los programadores ya que no se tienen que preocupar en que plataforma se utilizarán sus aplicaciones, solo enfocarse en la solución de la misma. Java realiza a través de la Maquina Virtual esta función, esta es una especie de puente entre el programa desarrollado y el sistema operativo donde se está corriendo.

La Máquina Virtual de Java (MVJ) genera un código conocido como "bytecode" que está a medio camino entre el programa y el código de máquina, que es el propio de cada hardware, es

este el que se entiende con las diferentes plataformas en lo que se ejecutan las aplicaciones. Esta portabilidad conseguida no está exenta de errores, ya que se presentan pequeñas inconsistencias cuando se cambia de plataforma, teniendo que hacer ciertas depuraciones, aún así Java es utilizado con gran éxito en programas para servidores o en servicios web y en dispositivos móviles.

1.2.7.2.3 Recolector de Basura

Java tiene un recolector de basura que cada cierto tiempo libera la memoria ocupada por objetos que ya no se van a utilizar más. Para realizarlo, el recolector se barre por la memoria marcando los objetos que han sido utilizados o referenciados, una vez terminado borra todos los que no estén marcados. Hay veces que igual se producen fugas de memoria ya que se hace referencia a objetos que ya no se están utilizando, lo que no es detectado por el recolector de basura. Básicamente su funcionalidad radica en que se realiza de manera casi invisible para el desarrollador, en otros lenguajes se realiza el mismo trabajo pero de manera manual, lo que causaba que a veces se tenga inestabilidad en los sistemas o fugas grandes de memoria al no liberar los espacios correctos.

Una desventaja del recolector de basura es que a veces retiene la memoria a liberar más del tiempo necesario y también que al realizar su trabajo el recolector demora cierto lapsos que pueden producir pausas lo que no puede permitirse en sistemas de tiempo real.

1.3 HERRAMIENTAS

1.3.1 NetBeans

NetBeans es un proyecto de código abierto que ha tenido mucho éxito con un gran número de usuarios y con más de 100 socios, el proyecto fue desarrollado en un principio por Sun MicroSystems en junio del 2000 y hasta el día de hoy sigue siendo el principal patrocinador de NetBeans.

NetBeans es un entorno de desarrollo integrado IDE (*Integrated Development Environment*), está escrito en el lenguaje Java. Cuanta con un conjunto de componentes de software que se llama módulos. Los módulos se pueden desarrollar independientemente lo que permite ir

agregándole nuevos módulos cuando sea necesario, además del mantenimiento es mucho más fácil cuando se lo realiza individual.

Características de NetBeans IDE

- Sirve para casi todos los lenguajes de programación pudiendo utilizar para escribir, compilar, depurar y ejecutar aplicaciones
- Soporta el desarrollo de casi todos los tipos de aplicación en Java (J2SE, web y aplicaciones móviles)
- Es un producto libre y gratuito sin restricciones de uso tanto comercial y no comercial. La versión 6.1 está bajo la licencia CDDL y la GPL
- La versión 6.1 tiene aproximadamente 1'990.915 líneas de código de las cuales el 99.19% pertenecen al lenguaje de Java.

Requisitos mínimos de hardware de NetBeans IDE 6.1

- **Microsoft Windows XP Professional SP3/Vista SP1/Windows 7 Professional:**
  Procesador: Intel Pentium III o equivalente a 800 MHz
  Memoria: 512 MB
  Espacio en disco: 750 MB de espacio libre en el disco

- **Ubuntu 9.10:**
  Procesador: Intel Pentium III o equivalente a 800 MHz
  Memoria: 512 MB
  Espacio en disco: 650 MB de espacio libre en el disco

- **Solaris OS versión 10 (SPARC):**
  Procesador: UltraSPARC II a 450 MHz
  Memoria: 512 MB
  Espacio en disco: 650 MB de espacio libre en el disco

- **Solaris OS versión 10 (x86/x64 Platform Edition):**
  Procesador: AMD Opteron serie 1200 a 1,8 GHz
  Memoria: 512 MB
  Espacio en disco: 650 MB de espacio libre en el disco

- **Macintosh OS X 10.5 Intel:**
  Procesador: Intel Dual-Core (32 o 64 bits)
  Memoria: 512 MB
  Espacio en disco: 650 MB de espacio libre en el disco

1.3.2 MySql

MySql es un sistema de gestión de base de datos relacional, se promociona como el más usado en el mundo del software libre, con más de 6 millones de instalaciones. Su aprobación se debe a que tiene una gran cantidad de librerías y herramientas para que su uso sea rápido y fácil. .

MySql se encuentra bajo la licencia GPL (GNU) y su distribución es gratuita, se diferencia en que su mantenimiento y desarrollo está bajo una empresa de Suecia, MySQL AB. Tienen también una versión comercial, con la única diferencia del soporte técnico y la posibilidad de integrarlo con algún software propietario.

Las características principales del MySql son:

- Con su implementación de multihilo aprovecha las ventajas de los sistemas multiprocesador.
- Mantiene un buen nivel de seguridad con gestión de usuarios y de contraseñas.
- Tiene una gran portabilidad entre los sistemas.
- Las columnas que maneja, soportan gran cantidad de tipos de datos.
- Si interfaz de programación maneja varios lenguajes como C, C++, PHP, Java, etc.

CAPITULO II:

En este capítulo se analizará la situación en la que se encuentra la institución, Esto se logrará con datos y documentos proporcionados por el rector de la Escuela "Teodoro Wolf N°72". A partir de estos se establecerán objetivos generales y específicos que se deben cumplir mediante un plan de desarrollo con estimaciones de lo que se va a implementar en que tiempo se lo realizará.

2.1 DESCRIPCIÓN DEL PROBLEMA

Una vez observado la manera de laborar en la Escuela Fiscal Mixta "Teodoro Wolf #72", se pudieron encontrar deficiencias en uso de recursos, como tiempo y materiales. Así también tareas repetitivas que fácilmente se podían automatizar con un sistema computarizado.

Los profesores escriben las notas y asistencias de los alumnos manualmente, y a su vez cada fin de bimestre se tiene que hacer cálculos para promedios y demás.

En el proceso de matriculación, la toma de datos es manualmente y da como resultado largos períodos de trabajo, que pueden ser aprovechados para actividades curriculares de aprendizaje de los alumnos. Como los datos son tomados en diferentes tiempos y de diferentes fuentes (alumnos, padres, profesores) estos contienen inconsistencias que son extendidas a todos los documentos que tiene la escuela, es decir, estos errores se presentan en los kardex, también en credenciales de alumnos, como en sus libretas de notas. Se deben mandar a imprimir plantillas de libretas, nóminas de alumnos, padres de familia y materias que posteriormente son llenadas manualmente.

2.2 OBJETIVOS ESTABLECIDOS

Las metas que se han planteado para resolver el problema que se encontró en la institución.

General

- Diseñar, desarrollar e implementar un sistema académico en Escuela Fiscal Mixta "Teodoro Wolf #72" de San Lorenzo – Manta - Manabí

Especifico

- Automatizar del registro de matrículas de los estudiantes.
- Crear un módulo para el manejo de calificaciones estudiantiles con sus respectivas materias.
- Imprimir documentos de la escuela como carnets, kardex y libretas de calificaciones.
- Registrar actividades de padres de familia.
- Integrar toda la información en una sola base de datos.

2.3 LEVANTAMIENTO DE REQUERIMIENTOS

Constan las funciones que va a realizar el sistema, en base a los requerimientos que tiene la unidad educativa.

- **F1: El sistema será capaz manejar de perfiles para los diferentes tipos de usuarios.**

  F1.1: Ingreso y contraseña para usuarios.

- **F2: El sistema será capaz de manejar el registro de los estudiantes.**

  F2.1: Ingresar información de estudiantes.

  F2.2: Eliminar datos de estudiantes.

  F2.3: Actualizar la información de los estudiantes, cuando este pase de año lectivo o exista cualquier cambio de información personal.

  F2.4: Consultar estudiantes.

  F2.4.1: Consulta general.

  F2.4.2: Consulta por parámetro**.**

  F2.4.2.1 Consulta por curso.

  F2.4.2.1 Consulta por código.

- **F3: El sistema será capaz de administrar la información de los profesores de la unidad educativa.**

  F3.1: Ingresar información de profesores.

  F3.2: Eliminar los datos de los profesores, en caso de que separarse de la institución.

  F3.3: Actualizar la información de los profesores, cuando exista alguna variante en los datos.

  F3.4: Consultar datos sobre profesores.

  F3.4.1: Consultarde manera general.

  F3.4.2: Consultar por cédula**.**

- **F4: En esta funcionalidad se podrá administrar la información de los padres de los alumnos**.

  F4.1: Ingresar los padres del alumno.

  F4.2: Eliminar los información de los padres.

  F4.3: Actualizar los datos de los representantes, en caso de alguna variante de la información.

  F4.4: Consultar sobre un padre de familia.

  F4.4.1: Consultar de manera general.

  F4.4.2: Consultar por código.

- **F5: El sistema será capaz de administrar la información de las materias que se imparten en la unidad educativa.**

  F5.1: Ingresar una materia.

  F5.2: Eliminar una materia, en caso que no se vuelva a impartir.

  F5.3: Actualizar los datos de una materia, en caso de que exista una variación en la información de la misma.

  F5.4: Consultar sobre una materia.

  F5.4.1: Consultar de manera general.

  F5.4.2: Consultar por parámetro.

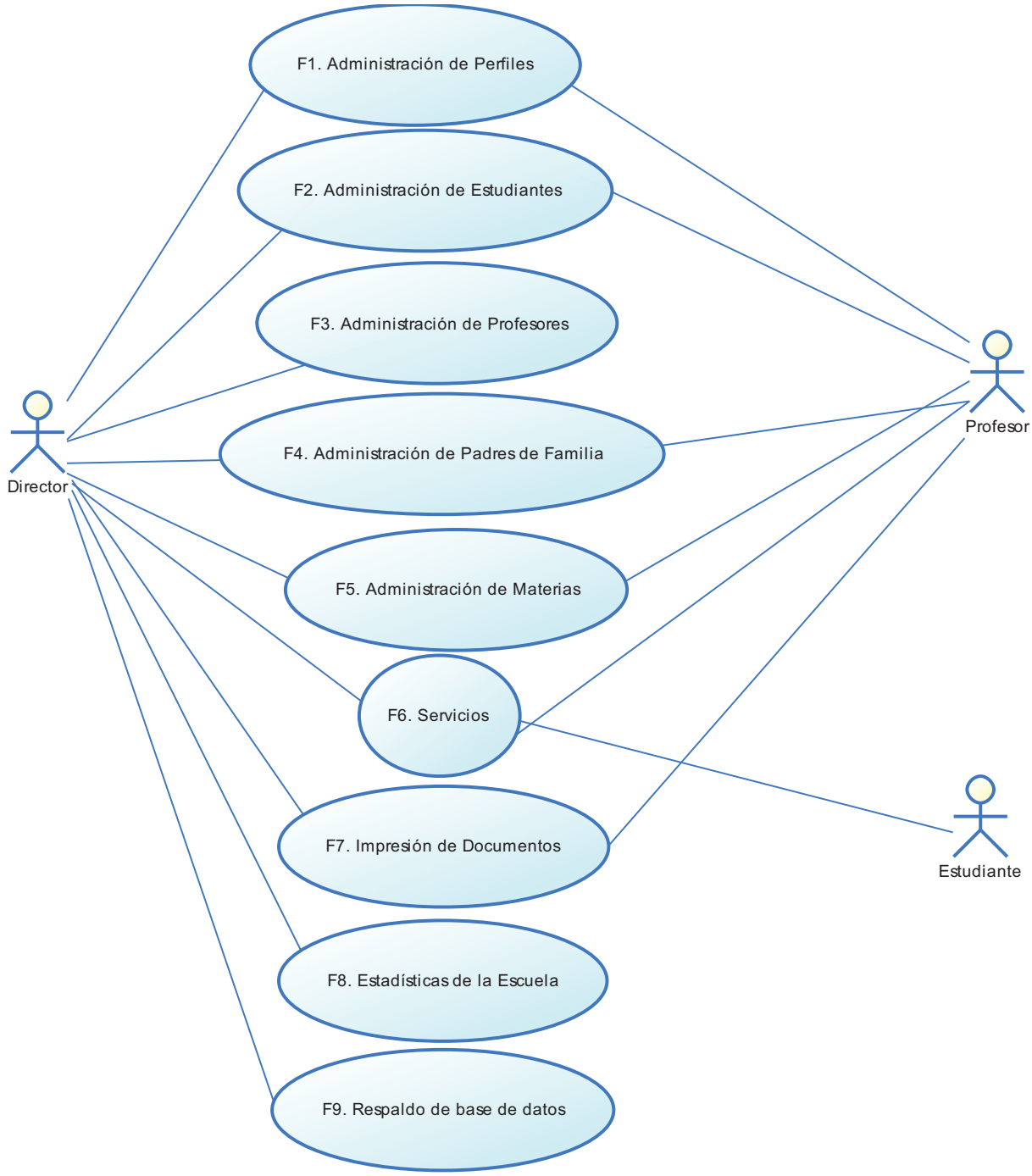
- **F6: El sistema realizará diferentes servicios para la unidad educativa.**
  - **F6.1:** Generar una libreta de calificaciones.
  - **F6.2:** Generar nuevo año escolar.

- **F7: El sistema será capaz de imprimir los documentos de la escuela.**
  - **F7.1:** Imprimir libreta de calificaciones.
    - **F7.1.1:** Imprimir por curso.
    - **F7.1.2:** Imprimir por alumno.
  - **F7.2:** Imprimir kardex estudiantes.
    - **F7.2.1:** Imprimir por curso.
    - **F7.2.2:** Imprimir por alumno.
  - **F7.3:** Imprimir carnets estudiantes.
    - **F7.3.1:** Imprimir por curso.
    - **F7.3.2:** Imprimir por alumno.
  - **F7.4:** Imprimir los horarios de los cursos.
    - **F7.4.1:** Imprimir por curso.
    - **F7.4.2:** Imprimir por alumno.
- **F8: El sistema manejará datos estadísticos de los diferentes aspectos de la unidad educativa.**
  - **F8.1:** Generar listado de abanderados.
  - **F8.2:** Generar listado de horas por profesor.
  - **F8.3**: Generar estadística de notas por curso.
  - **F8.4:** Generar estadística de notas por materias.
- **F9: El sistema administrará respaldos de la base de datos.**
  - **F9.1** Respaldar la base de datos por seguridad.

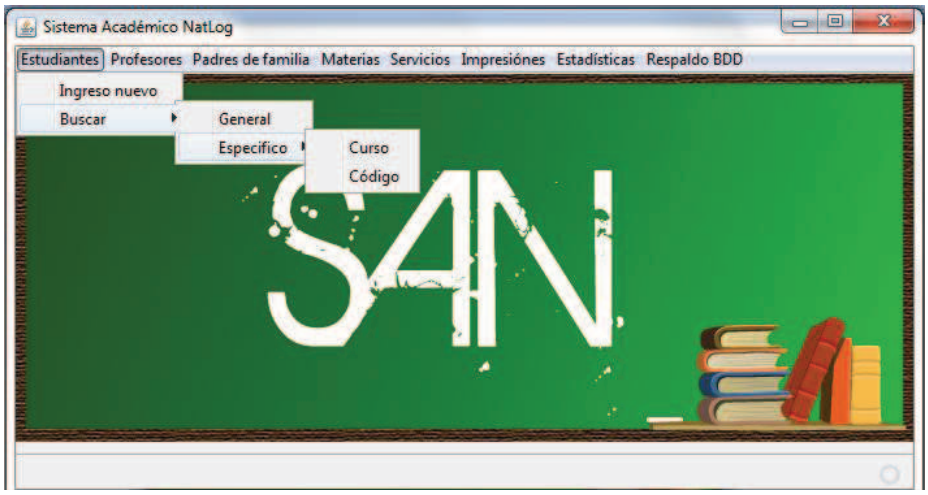
2.4 ANÁLISIS DE REQUERIMIENTOS

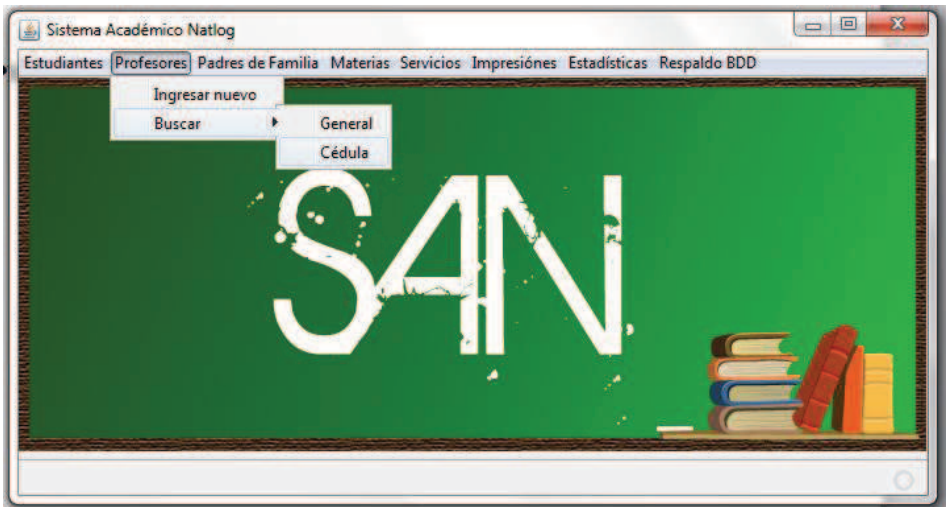
Se va a realizar un análisis general de los requerimientos que fueron pedidos por la unidad educativa.

2.4.1 Diagramas de casos de uso general (nivel 0)



2.5 PROTOTIPO DEL SISTEMA ACADÉMICO

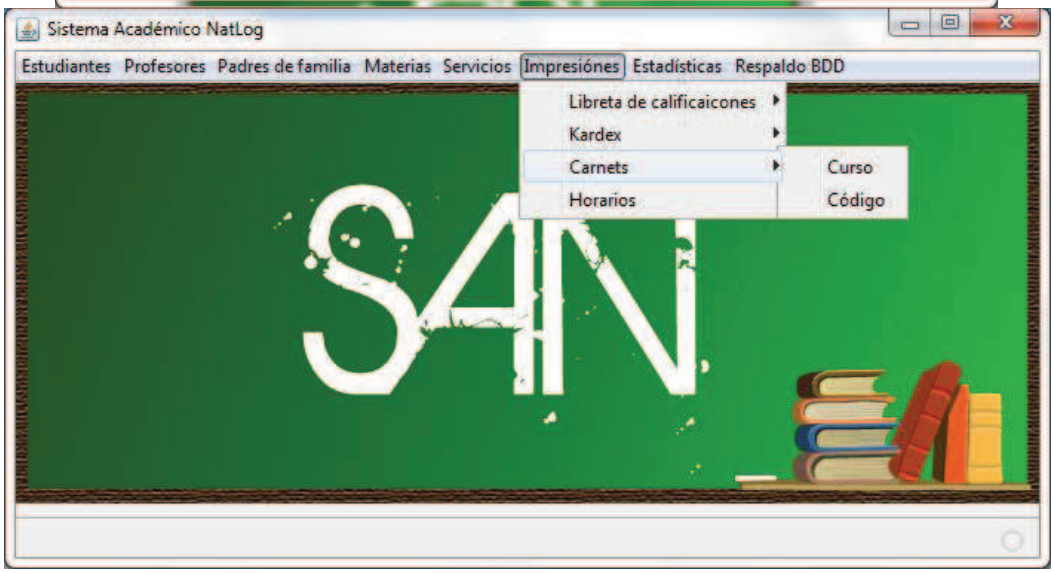







Sistema Académico Natlog
Estudiantes Profesores Padres de Familia Materias Servicios Impresiónes Estadísticas Respaldo BDD
Ingresar nueva
Buscar
General
Código
SAN


Sistema Académico NatLog
Estudiantes Profesores Padres de familia Materias Servicios Impresiónes Estadísticas Respaldo BDD
Libreta de Calificaciones
Nuevo año escolar
SAN


Sistema Académico NatLog
Estudiantes Profesores Padres de familia Materias Servicios Impresiónes Estadísticas Respaldo BDD
Libreta de calificaicones
Kardex
Carnets
Horarios
Curso
Código
SAN


Sistema Académico Natlog
Estudiantes
Profesores
Padres de Familia
Materias
Servicios
Impresiónes
Estadísticas
Respaldo BDD
Horas por profesor
Notas por curso
Notas por materias
SAN


Sistema Académico Natlog
Estudiantes
Profesores
Padres de Familia
Materias
Servicios
Impresiónes
Estadísticas
Respaldo BDD
Respaldo actual
SAN

2.6 PLANIFICACIÓN DE ENTREGAS

Se presenta una tabla con las estimaciones de líneas de código y de tiempo, de las funcionalidades divididas en 3 ciclos.

TSPi Strategy Form: Form STRAT

		Cycle LOC			Cycle Hours		
Reference	Functions	1	2	3	1	2	3
F1	**Administración de Perfiles**						
F1.1	Validación de Usuario	50			3		
F2.	**Administración de Estudiantes**						
F2.1	Ingresar estudiante	80			4		
F2.2	Eliminar estudiante	30			1.8		
F2.3	Actualizar estudiante	40			2		
F2.4	Consultar estudiante						
F2.4.1	Consulta general.	50			3		
F2.4.2	Consulta por parámetro**.**						
F2.4.2.1	Consulta por curso.	30			1.8		
F2.4.2.1	Consulta por código.	30			1.8		
F3	**Administración de Profesores**						
F3.1	Ingresar profesor	80			4		
F3.2	Eliminar profesor	30			1.8		
F3.3	Actualizar profesor	40			2		
F3.4	Consultar profesor						
F3.4.1	Consulta general	30			1.8		
F3.4.2	Consulta por cédula	30			1.8		
F4	**Administración de Padres de Familia**						
F4.1	Ingresar padre de familia	80			3		

F4.2	Eliminar padre de familia	30			1.8		
F4.3	Actualizar padre de familia	60			2.5		
F4.4	Consultar padre de familia						
F4.4.1	Consulta general	30			1.8		
F4.4.2	Consulta por código	30			1.8		
F5	**Administración de Materias**						
F5.1	Ingresar materia	60			2.5		
F5.2	Eliminar materia	20			1.5		
F5.3	Actualizar materia	35			2.2		
F5.4	Consultar materia						
F5.4.1	Consulta general	30			1.8		
F5.4.2	Consulta por código	30			1.8		
F6	**Servicios**						
F6.1	Generar libreta de calificaciones		180			10	
F6.2	Generar nuevo año escolar		130			8	
F7	**Impresión de Documentos**						
F7.1	Imprimir libreta de calificaciones.						
F7.1.1	Imprimir por curso.			80			4
F7.1.2	Imprimir por alumno.			60			3
F7.2	Imprimir kardex estudiantes.						
F7.2.1	Imprimir por curso.			80			4
F7.2.2	Imprimir por alumno.			60			3
F7.3	Imprimir carnets estudiantes.						
F7.3.1	Imprimir por curso.			80			4
F7.3.2	Imprimir por alumno.			60			3
F7.4	Imprimir los horarios de los cursos.			60			3
F8	**Estadísticas de la Escuela**						
F8.1	Generar listado de horas por		35			2.5	

	profesor.						
F8.2	Generar estadística de notas por curso.		35			2.5	
F8.3	Generar estadística de notas por materias.		35			2.5	
F9	**Respaldo de base de datos**						
F9.1	Respaldo puntual de base de datos		50			2.8	
Total		925	595	480	49,5	36,3	24

CAPITULO III: DESARROLLO DEL SISTEMA ACADÉMICO

En este capítulo se delimitara el trabajo en partes, se realizaran diagramas y explicara paso a paso lo que se lo va a realizar mediante las herramientas y metodologías que se han explicado en capítulos anteriores para poder desarrollar el aplicativo para la Escuela Fiscal Mixta "Teodoro Wolf N°72".

3.1 ESTRUCTURA DE REQUERIMENTOS

El proyecto se va a dividir en iteraciones incrementales que cubrirán todas las funcionalidades del programa. Están distribuidas de la siguiente forma:

Iteración 1
Administración de Perfiles
Administración de Estudiantes
Administración de Profesores
Administración de Padres de Familia
Administración de Materias

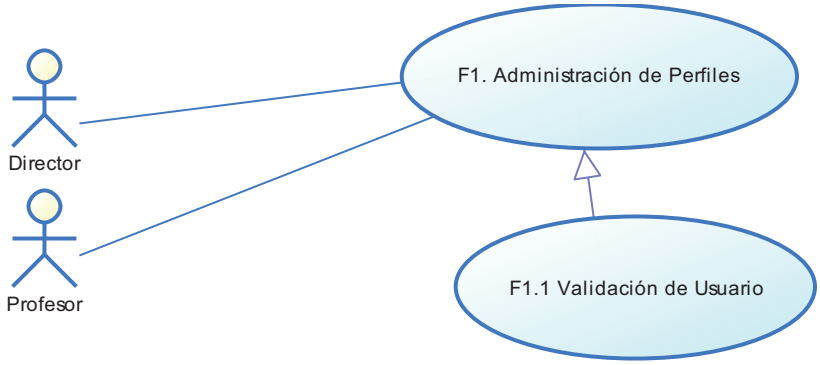
Iteración 2
Servicios
Estadísticas de la Escuela
Respaldo de base de datos

Iteración 3
Impresión de Documentos

3.1.1 Diagramas de casos de uso a detalle

Se va a realizar un análisis detallado uno a uno de los requerimientos que fueron pedidos por la unidad educativa.

- **F1. Administración de Perfiles**

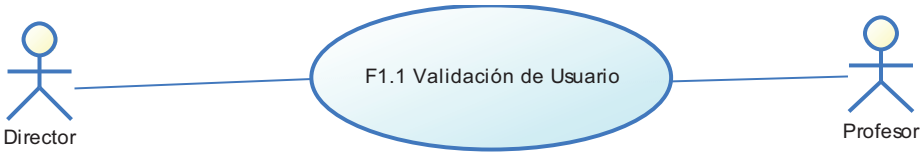


Detalle:

- **F1.1: Validación de Usuario**

A través de este caso uso aceden los actores al sistema.

Actores: Director y profesor



Descripción:

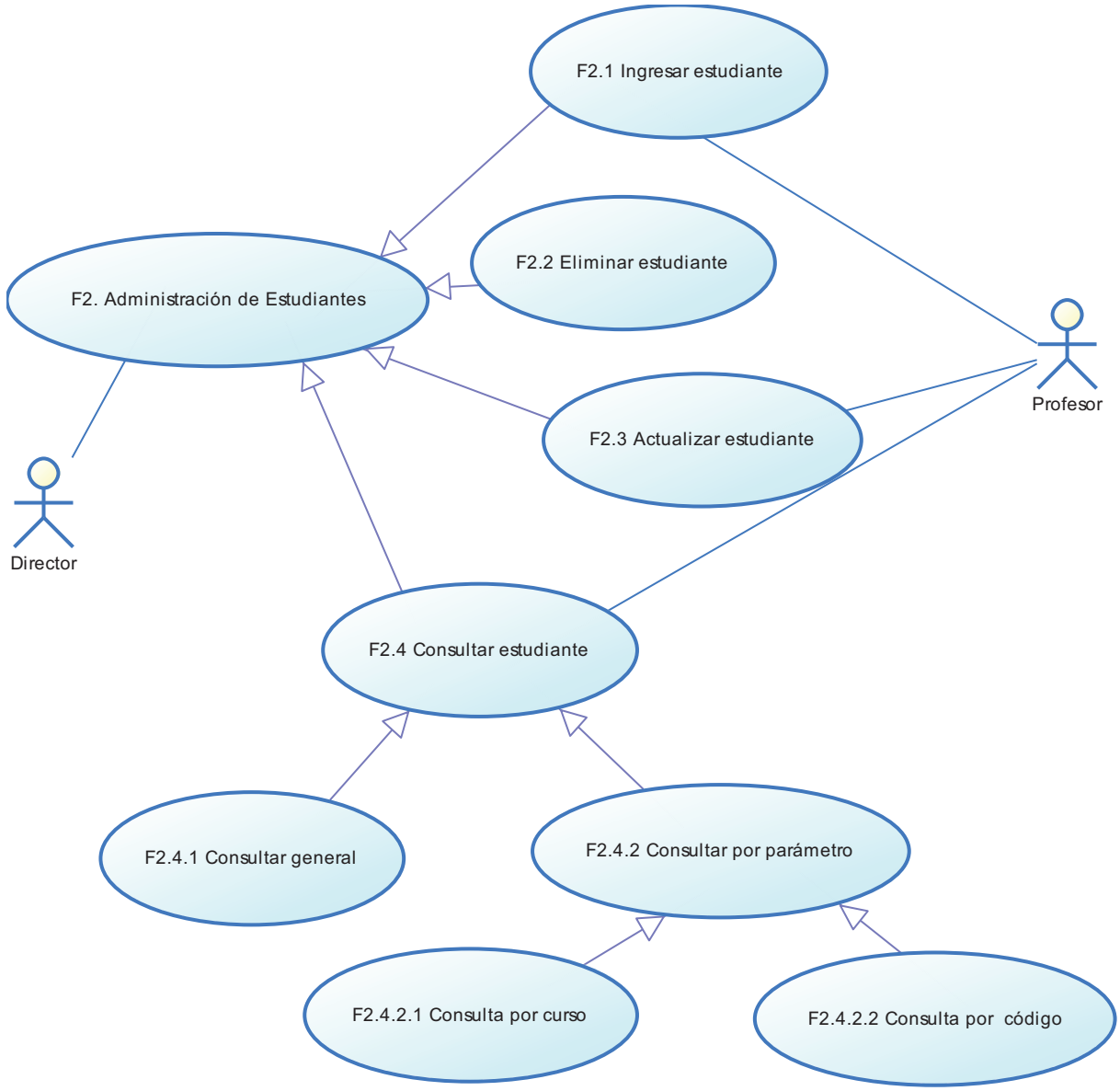
Paso	Actor	Paso	Sistema	Exep.
1	El director o profesor escogerá la opción de ingresar al sistema académico.	**2**	El programa desplegará la ventana de ingreso de usuario y contraseña.	
3	El director o profesor ingresará sus respectivas			

	credenciales, usuario y contraseña.			
4	El director o profesor da clic en el botón Ingresar	**5**	El programa mostrará las opciones que se han habilitado de acuerdo al perfil que se ha asignado a ese usuario	**E1** **E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

- **F2. Administración de Estudiantes**

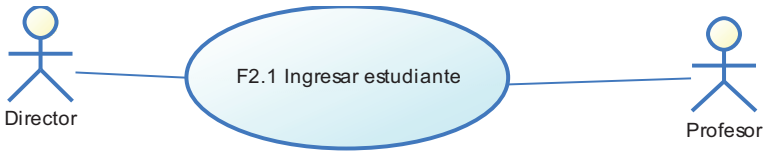


Detalle:

- **F2.1: Ingresar estudiante**

  A través de este caso uso se ingresa un nuevo estudiante al sistema.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de ingresar un nuevo estudiante.	**2**	El sistema muestra la ventana con el formulario del nuevo estudiante.	
3	El director o el profesor, ingresan los datos de estudiante.			
4	El director o el profesor, dan clic en el botón Guardar.	**5**	El programa muestra una ventana indicando que se han aceptado los cambios.	**E1** **E2**

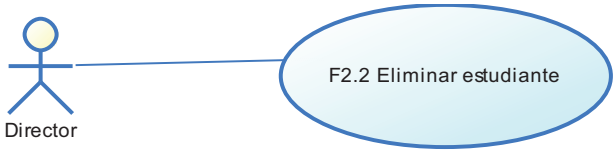
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Las credenciales ingresadas son incorrectas.	Ingresar datos de nuevo.

- **F2.2: Eliminar estudiante**

  A través de este caso uso se elimina un estudiante del sistema.

  Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción buscar estudiante.	**2**	El sistema muestra una ventana para el ingreso del código.	
3	El director ingresa el código del estudiante.			
4	El director da clic en el botón Aceptar.	**5**	El sistema verifica que el estudiante con el código ingresado este en la base de datos.	**E1** **E2**
		6	El sistema despliega todos los datos del estudiante.	**E1**
7	El director da clic en el botón Eliminar.	**8**	El sistema despliega una ventana con un mensaje preguntando si se confirma la eliminación.	
9	El director da clic en el botón Confirmar.	**10**	El sistema despliega una ventana con el mensaje de aceptación de los cambios	**E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos	Intentar nuevamente
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo

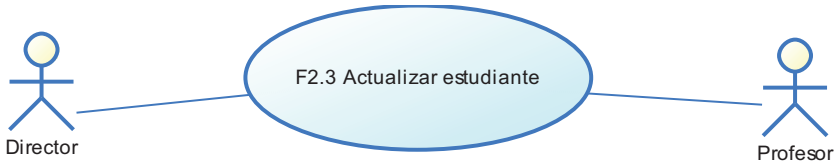
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F2.3 Actualizar estudiante**

A través de este caso uso se actualiza un estudiante del sistema.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escoge la opción buscar estudiante por código.	**2**	El sistema despliega una ventana de ingreso de código.	
3	El director o el profesor, ingresan el código del estudiante.			
4	El director o el profesor, dan clic en el botón Aceptar.	**5**	El sistema verifica que el estudiante con el código ingresado este en la base de datos.	**E1** **E2**
		6	El sistema despliega todos los datos del estudiante.	**E1**
7	El director o el profesor, dan clic en el botón Actualizar.	**8**	El sistema desbloquea los campos para que se puedan cambiar los datos.	
9	El director o el profesor cambian los datos que desea actualizar.			
10	El director o el profesor, dan clic en el botón Guardar.	**11**	El sistema despliega una ventana con el mensaje de aceptación de los cambios.	**E2**
		12	El sistema vuelve a bloquear los	

campos.

Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos	Intentar nuevamente
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo

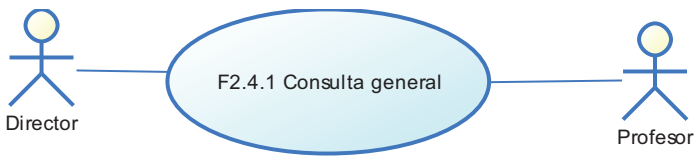
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F2.4.1 Consulta general**

  A través de este caso uso se consultan todos los estudiantes que se encuentran en el sistema.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción búsqueda general de estudiantes.	**2**	El sistema despliega la ventana con el listado de estudiantes.	**E1**

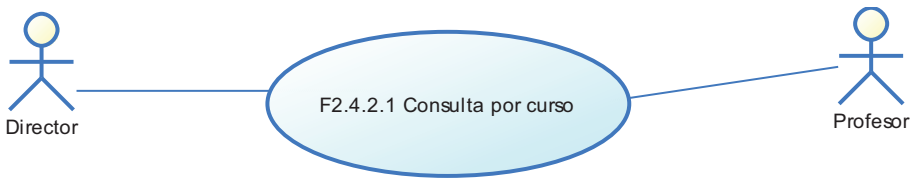
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F2.4.2.1 Consulta por curso**

  A través de este caso uso se consultan todos los estudiantes de un curso en particular.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción búsqueda de estudiantes por curso.	**2**	El sistema despliega una ventana con todos los cursos.	
3	El director o el profesor, escogen el curso que desea.			
4	El director o el profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega la ventana con el listado de estudiantes del curso	**E1**

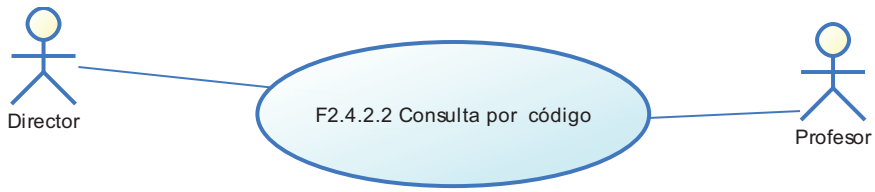
Excepción:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos	Intentar nuevamente

- **F2.4.2.1 Consulta por código**

A través de este caso uso se consulta a un estudiante específico.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escoge la opción búsqueda de estudiantes por código.	**2**	El sistema despliega una ventana para ingresar el código.	
3	El director o el profesor, escribe el código.			
4	El director o el profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega la ventana con los datos del estudiante	**E1** **E2**

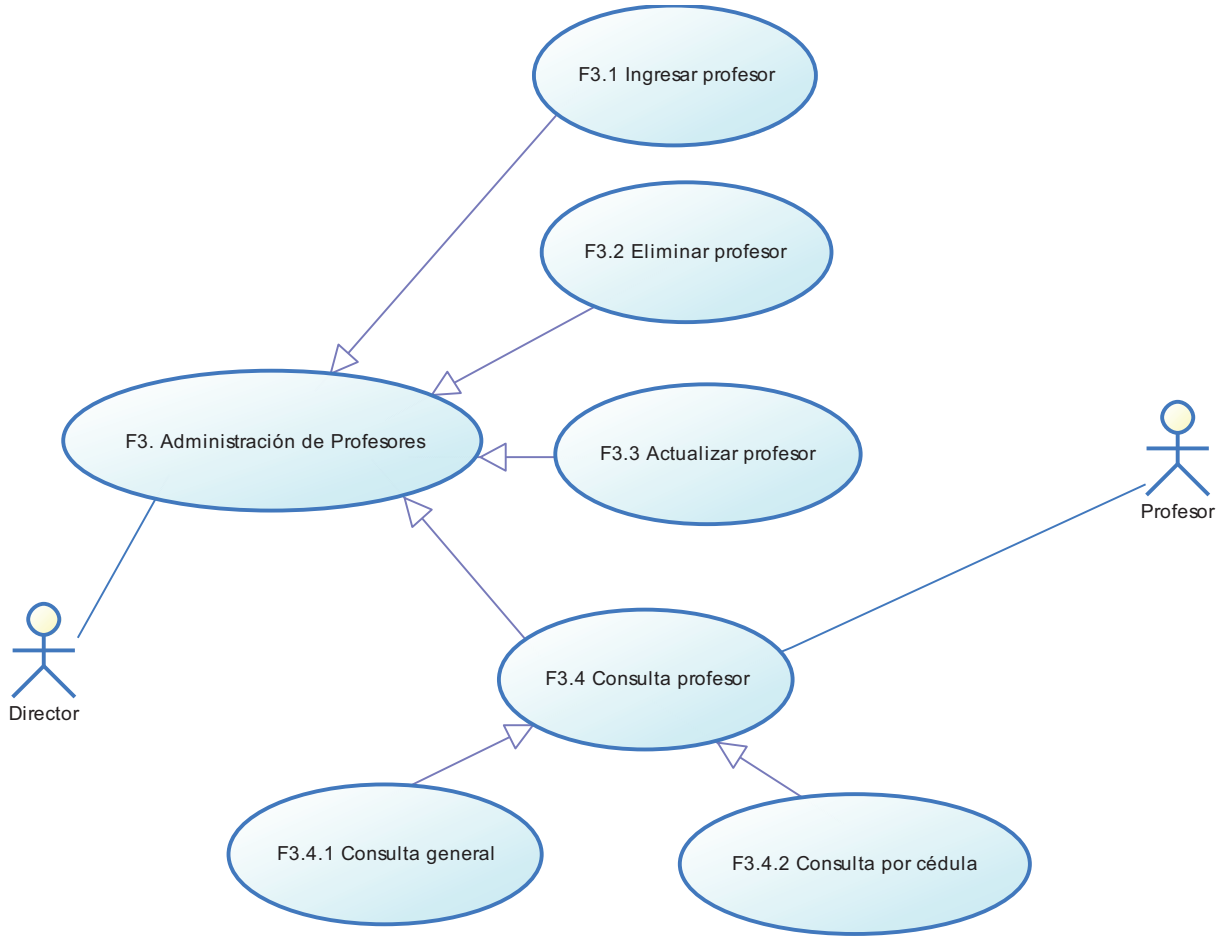
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos	Intentar nuevamente
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F3. Administración de Profesores**



Detalle:

- **F3.1 Ingresar profesor**

  A través de este caso uso se ingresa un nuevo profesor al sistema.

  Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción de ingresar un nuevo profesor.	**2**	El sistema muestra la ventana con el formulario del nuevo profesor.	
3	El director ingresa la cédula del nuevo profesor.			
4	El director da clic en el botón Aceptar.	**5**	El sistema verifica que esa cédula no esté siendo utilizada en la base de datos.	**E1**
6	El director ingresa el resto de los datos del profesor.			
7	El director da clic en el botón Guardar.	**8**	El programa mostrará una ventana indicando que se han aceptado los cambios.	**E2**

Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos	Intentar nuevamente
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo

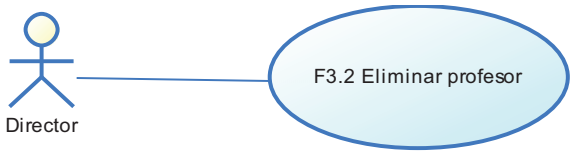
Flujo alterno:

Paso	Descripción
5	Si existe, verificar F3.2 y F3.3

- **F3.2 Eliminar profesor**

A través de este caso uso se elimina un profesor del sistema.

Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción buscar profesor.	**2**	El sistema despliega una ventana para ingresar la cédula.	
3	El director ingresa la cédula del profesor.			
4	El director da clic en el botón Aceptar.	**5**	El sistema verifica que el profesor con la cédula ingresada este en la base de datos.	**E1** **E2**
		6	El sistema despliega todos los datos del profesor.	**E1**
7	El director da clic en el botón Eliminar.	**8**	El sistema despliega una ventana con un mensaje preguntando si se confirma la eliminación.	
9	El director da clic en el botón Confirmar.	**10**	El sistema despliega una ventana con el mensaje de aceptación de los cambios	**E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F3.1

- **F3.3 Actualizar profesor**

A través de este caso uso se actualiza un profesor al sistema.

Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción buscar profesor.	**2**	El sistema despliega una ventana para ingresar la cédula.	
3	El director ingresa la cédula del profesor.			
4	El director da clic en el botón Aceptar.	**5**	El sistema verifica que el profesor con la cédula ingresada esté en la base de datos	**E1** **E2**
		6	El sistema despliega todos los datos del profesor.	**E1**
7	El director da clic en el botón Actualizar	**8**	El sistema desbloquea los campos para que se puedan cambiar los datos.	
9	El director cambia los datos que desea actualizar.			

10	El director da clic en el botón Guardar.	**11**	El sistema despliega una ventana con el mensaje de aceptación de los cambios.	**E2**
		11	El sistema vuelve a bloquear los campos.	

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

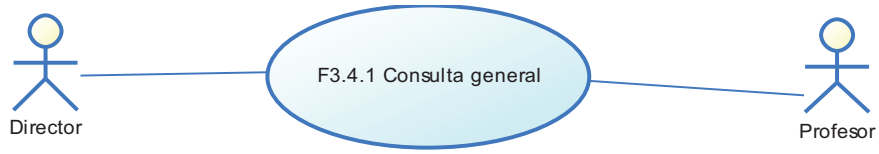
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F3.1

- **F3.4.1 Consulta general**

  A través de este caso uso se realiza una consulta general de los profesores del sistema.

Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o profesor, escogen la opción	**2**	El sistema despliega la ventana con el listado de profesores.	**E1**

	búsqueda general de profesores.

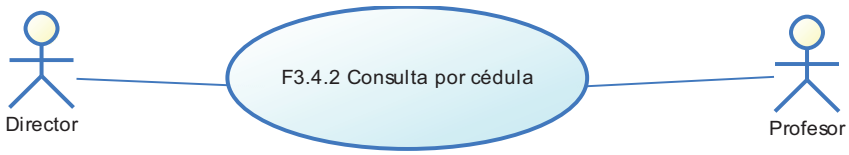
Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F3.4.2 Consulta por cédula**

A través de este caso uso se realiza una consulta de un profesor específico.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción búsqueda de profesores por parámetros.	**2**	El sistema despliega una ventana para ingresar la cédula.	
3	El director o el profesor, escriben la cédula del profesor.			
4	El director o el profesor, dan clic en botón Aceptar.	**5**	El sistema despliega la ventana con los datos del profesor.	**E1**

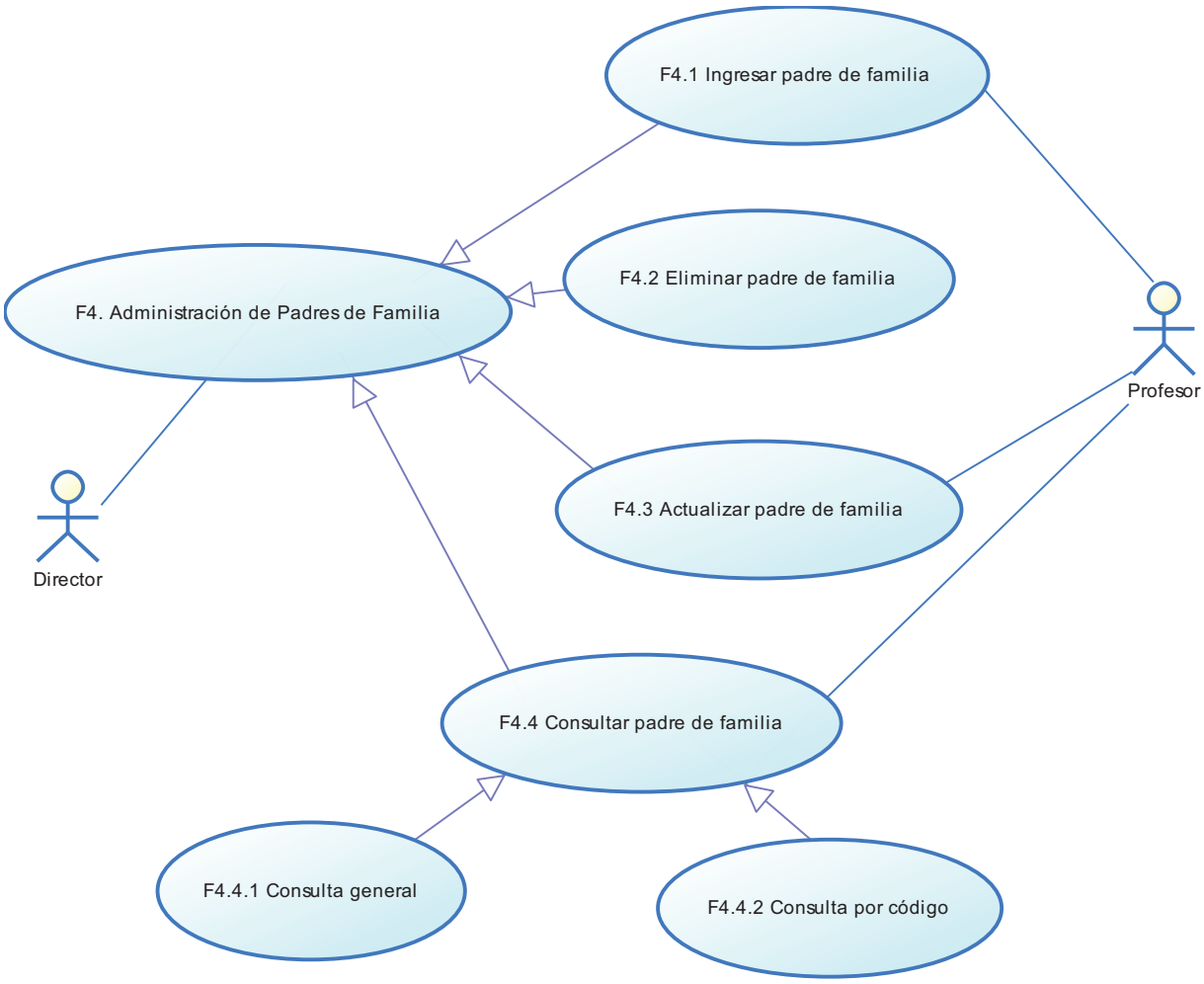
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F3.1

- **F4. Administración de Padres de Familia**

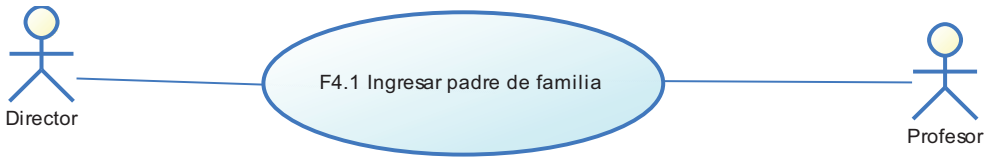


Detalle:

- **F4.1 Ingresar padre de familia**

  A través de este caso uso se ingresa un nuevo padre de familia

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de ingresar nuevo padre de familia.	**2**	El sistema despliega el formulario de nuevo padre de familia.	
3	El director o el profesor, ingresan los datos del nuevo padre de familia.			
4	El director o el profesor da clic en el botón Guardar.	**5**	El programa mostrará una ventana indicando que se han aceptado los cambios.	**E1** **E2**

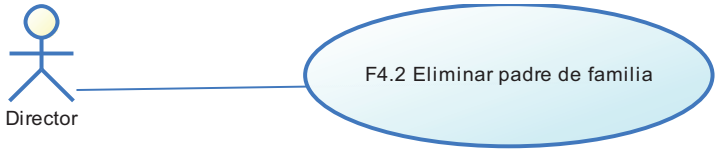
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E3	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

- **F4.2 Eliminar padre de familia**

  A través de este caso uso se elimina un padre de familia

Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción buscar padre de familia.	**2**	El sistema despliega una ventana para el ingreso del código.	
3	El director ingresa el código del padre de familia.			
4	El director da clic en el botón Aceptar.	**5**	El sistema verifica que el padre de familia con el código ingresado este en la base de datos.	**E1** **E2**
		6	El sistema despliega todos los datos del padre de familia.	**E1**
7	El director da clic en el botón Eliminar.	**8**	El sistema despliega una ventana con un mensaje preguntando si se confirma la eliminación.	
9	El director da clic en el botón Confirmar.	**10**	El sistema despliega una ventana con el mensaje de aceptación de los cambios	**E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Las credenciales ingresadas son incorrectas.	Ingresar datos de nuevo.

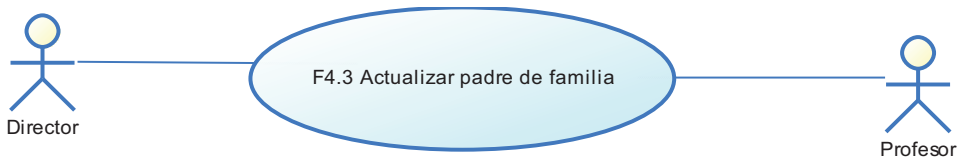
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F4.1

- **F4.3 Actualizar padre de familia**

A través de este caso uso se actualiza un padre de familia

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o profesor, escogen la opción buscar padre de familia	**2**	El sistema despliega una ventana para ingresar el código.	
3	El director o profesor, ingresan el código del padre de familia.			
4	El director o profesor, dan	**5**	El sistema verifica que el padre de	**E1**

	clic en el botón Aceptar.		familia con el código ingresado esté en la base de datos	**E2**
		6	El sistema despliega todos los datos del padre de familia.	**E1**
7	El director o profesor, dan clic en el botón Actualizar.	**8**	El sistema desbloquea los campos para que se puedan cambiar los datos.	
9	El director o profesor, cambian los datos que desea actualizar.			
10	El director o profesor, dan clic en el botón Guardar.	**11**	El sistema despliega una ventana con el mensaje de aceptación de los cambios.	**E2**
		12	El sistema vuelve a bloquear los campos.	

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

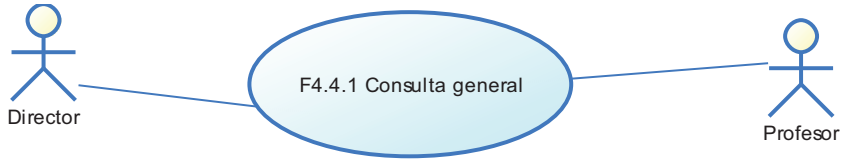
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F4.1

- **F4.4.1 Consulta general**

  A través de este caso uso se consultan todos los padres de familia.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escoge la opción búsqueda general de padres de familia.	2	El sistema despliega la ventana con el listado de los padres de familia.	E1

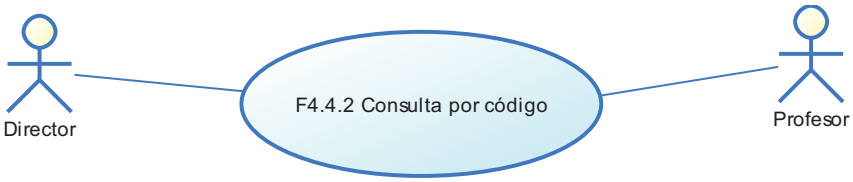
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F4.4.2 Consulta por código**

  A través de este caso uso se consultan todos los padres de familia.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción búsqueda de padres de familia por parámetros.	**2**	El sistema despliega una venta para ingresar el código del padre de familia.	
3	El director o el profesor, escriben el código del padre de familia.			
4	El director o profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega la ventana con los datos del padre de familia.	**E1** **E2**

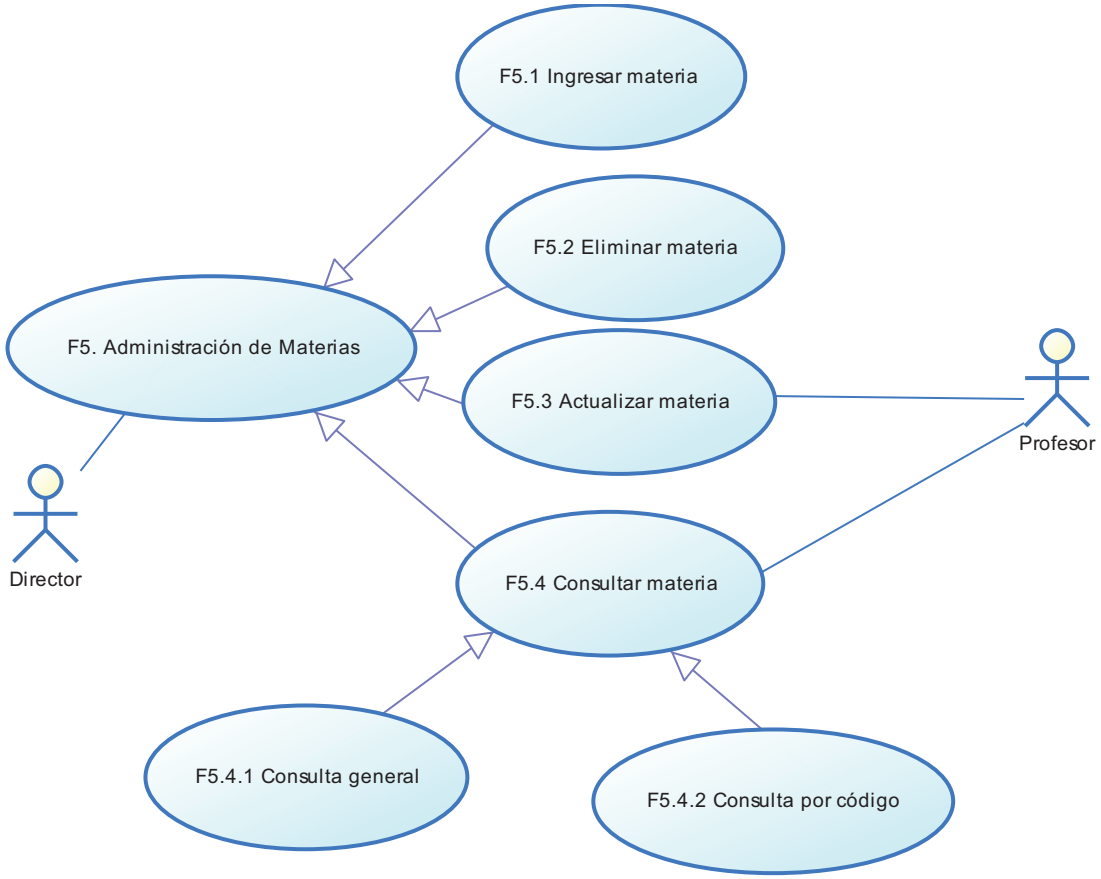
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F4.1

- **F5. Administración de Materias**



Detalle:

- **F5.1 Ingresar materia**

  A través de este caso uso se ingresa una nueva materia.

  Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción de ingresar una nueva materia.	**2**	El sistema muestra la ventana con el formulario de la nueva materia.	
3	El director ingresa el resto de los datos de la materia.			
4	El director da clic en el botón Guardar.	**5**	El programa mostrará una ventana indicando que se han aceptado los cambios.	**E1** **E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

- **F5.2 Eliminar materia**

A través de este caso uso se elimina una materia.

Actores: Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción buscar materia.	**2**	El sistema despliega una ventana para el ingreso del código.	
3	El director ingresa el código de la materia.			
4	El director o profesor, dan clic en el botón Aceptar.	**5**	El sistema verifica que la materia con el código ingresado este en la base de datos.	**E1** **E2**
		6	El sistema despliega todos los datos de la materia.	**E1**
7	El director da clic en el botón Eliminar.	**8**	El sistema despliega una ventana con un mensaje preguntando si se confirma la eliminación.	
9	El director da clic en el botón Confirmar.	**10**	El sistema despliega una ventana con el mensaje de aceptación de los cambios	**E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

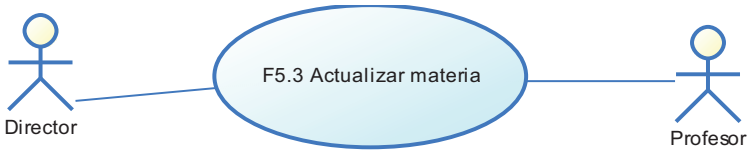
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F5.1

- **F5.3 Actualizar materia**

A través de este caso uso se actualiza una materia.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o profesor, escogen la opción buscar materia.	**2**	El sistema despliega una ventana para el ingreso del código.	
3	El director o profesor, ingresan el código de la materia.			
4	El director o profesor, dan clic en el botón Aceptar.	**5**	El sistema verifica que el código esté ingresado en la base de datos	**E1** **E2**
		6	El sistema despliega todos los datos de la materia.	**E1**
7	El director o profesor, dan clic en el botón Actualizar.	**8**	El sistema desbloquea los campos para que se puedan cambiar los datos.	
9	El director o profesor,			

	cambian los datos que desea actualizar.			
10	El director o profesor, dan clic en el botón Guardar.	**11**	El sistema despliega una ventana con el mensaje de aceptación de los cambios.	**E2**
		12	El sistema vuelve a bloquear los campos.	

Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

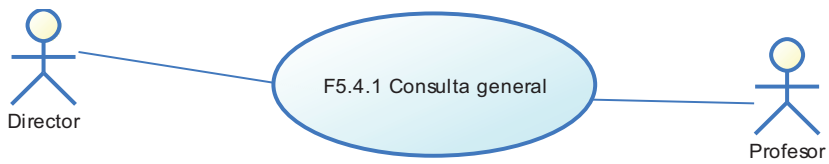
Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F5.1

- **F5.4.1 Consulta general**

  A través de este caso uso se consultan todas las materias.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escoge la opción búsqueda general de materias	**2**	El sistema despliega la ventana con el listado de las materias.	**E1**

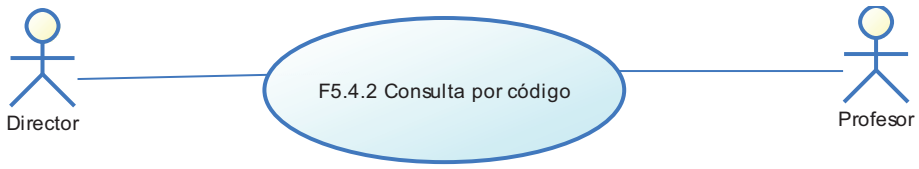
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F5.4.2 Consulta por código**

  A través de este caso uso se consulta una materia específica.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción búsqueda de materia por parámetros.	**2**	El sistema despliega una ventana para el ingreso del código de materia.	
3	El director o el profesor, escriben el código de la			

	materia.			
4	El director o profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega la ventana con los datos de la materia.	**E1** **E2**

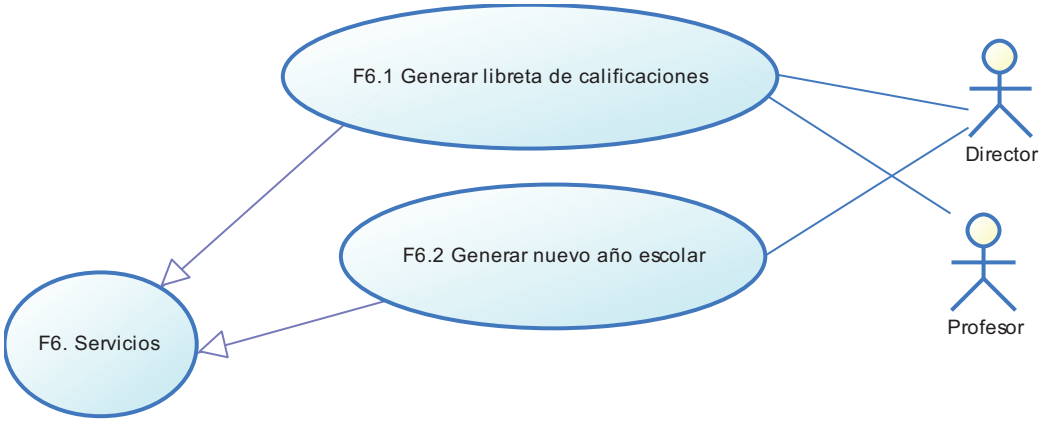
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F5.1

- **F6. Servicios**

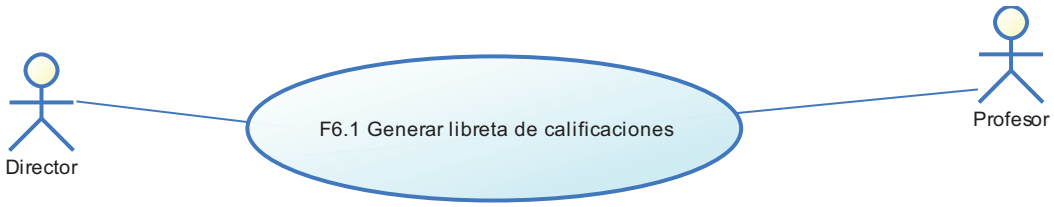


Detalle:

- **F6.1 Generar libreta de calificaciones**

  A través de este caso uso se genera la libreta de calificaciones.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción generar libreta de calificaciones	**2**	El sistema despliega una ventana para el ingreso del código.	
3	El director o el profesor, escriben el código del			

	alumno que se va realizar la libreta de calificaciones			
4	El director o profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega una ventana, con campos parcialmente llenos, dependiendo de lo que se lea en la base de datos.	**E1** **E2**
6	El director o el profesor, llenan los campos de la libreta de calificaciones.			**E2**
7	El director o el profesor, dan clic en el botón de Guardar	**8**	El sistema responde con un mensaje de aceptación de cambios	**E1** **E2**

Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F6.2 Generar nuevo año escolar**

A través de este caso uso se cambia a un nuevo año escolar.

Actores: Director



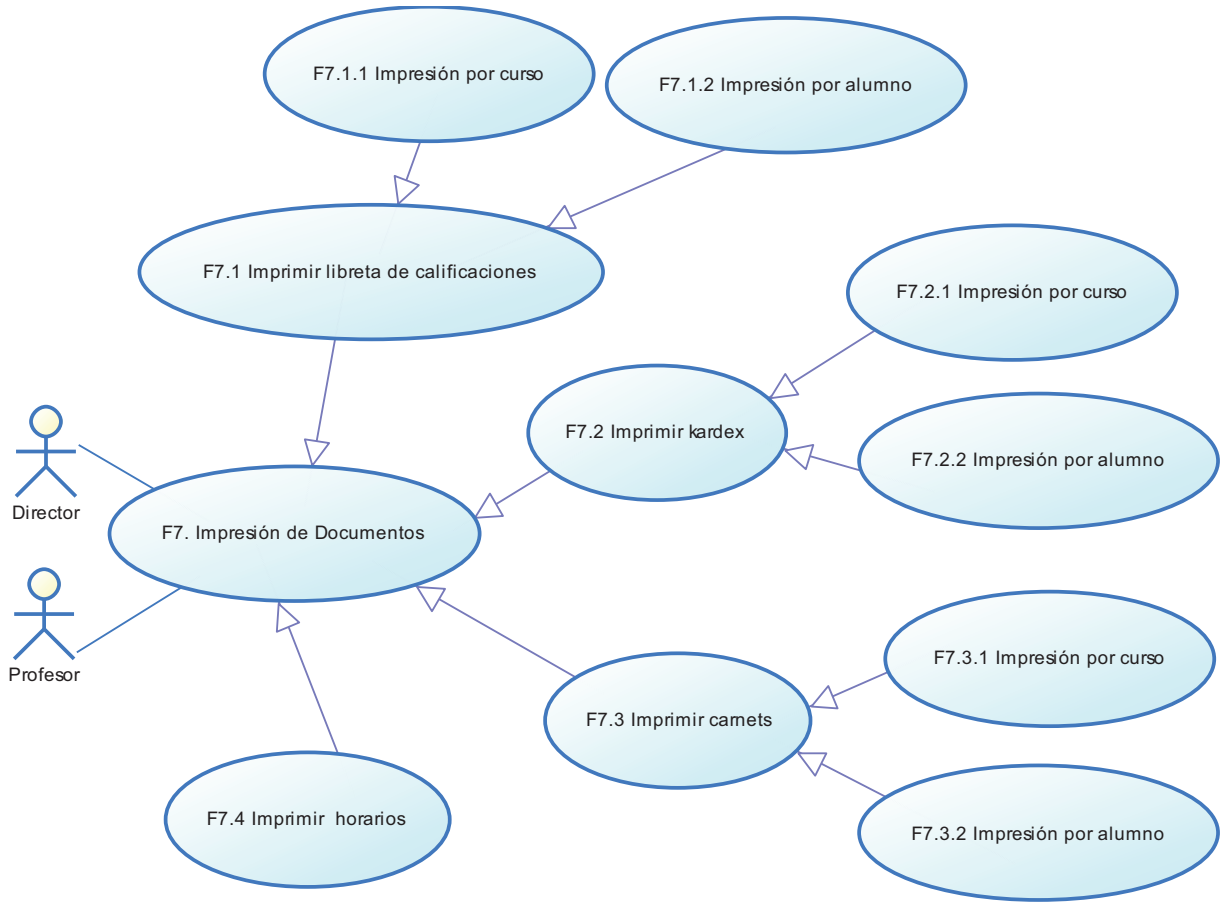
Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción de generar un nuevo año escolar	**2**	El sistema despliega una ventana con un mensaje para que se confirme el requerimiento.	
3	El director confirma que desea crear un nuevo año escolar.	**4**	El sistema muestra un mensaje de aceptación de cambios.	**E1**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

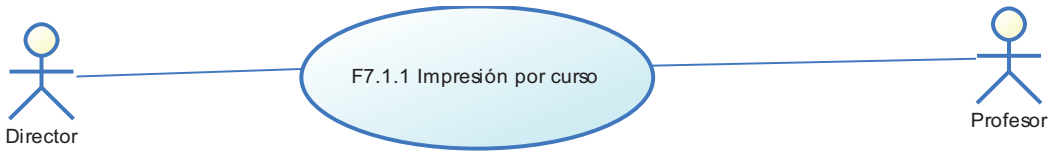
- **F7. Impresión de Documentos**



- **F7.1.1 Impresión por curso**

  A través de este caso uso se imprime las libretas de calificaciones de un curso entero.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir libreta de calificaciones por curso.	**2**	El sistema despliega un listado de cursos.	
3	El director o el profesor, escogen el curso de la lista.			
4	El director o el profesor, dan clic en el botón Imprimir.	**5**	El sistema envía a imprimir la información.	**E1** **E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Error de impresión.	Intentar nuevamente.

- **F7.1.2 Impresión por alumno**

  A través de este caso uso se imprime la libreta de calificaciones de un solo alumno.

  Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir libreta de calificaciones por alumno.	**2**	El sistema muestra una ventana para ingresar el código.	
2	El director o el profesor, ingresan el código del estudiante.			
3	El director o el profesor, dan clic en el botón Aceptar.	**4**	El sistema despliega la libreta de calificaciones del alumno.	**E1** **E2**
5	El director o el profesor, dan clic en el botón Imprimir.	**6**	El sistema envía a imprimir la información.	**E3**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.
E3	Error de impresión.	Intentar nuevamente.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F7.2.1 Impresión por curso**

  A través de este caso uso se imprime los kardex de calificaciones de un curso entero.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir kardex por curso.	**2**	El sistema despliega un listado de cursos.	
3	El director o el profesor, escoge el curso de la lista.			
4	El director o el profesor, dan clic en el botón Imprimir.	**5**	El sistema envía a imprimir la información.	**E1** **E2**

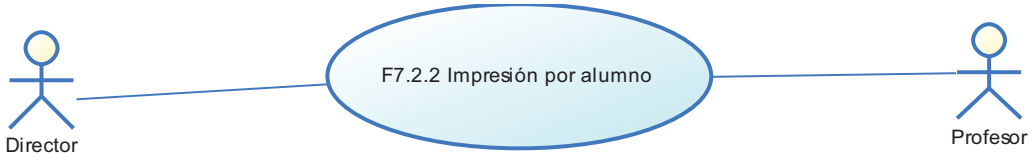
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Error de impresión.	Intentar nuevamente.

- **F7.2.2 Impresión por alumno**

  A través de este caso uso se imprime el kardex de calificaciones de un alumno específico.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir kardex por alumno.	**2**	El sistema despliega una ventana para ingreso del código.	
3	El director o el profesor, ingresan el código del estudiante.			
4	El director o el profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega el kardex del alumno.	**E1** **E2**
6	El director o el profesor, dan clic en el botón Imprimir.	**7**	El sistema envía a imprimir la información.	**E3**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.
E3	Error de impresión.	Intentar nuevamente.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F7.3.1 Impresión por curso**

A través de este caso uso se imprime los carnets de un curso entero.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir carnets por curso.	**2**	El sistema despliega un listado de cursos.	
3	El director o el profesor, escogen el curso de la lista.			
4	El director o el profesor, dan clic en el botón Imprimir.	**5**	El sistema envía a imprimir la información.	**E1** **E2**

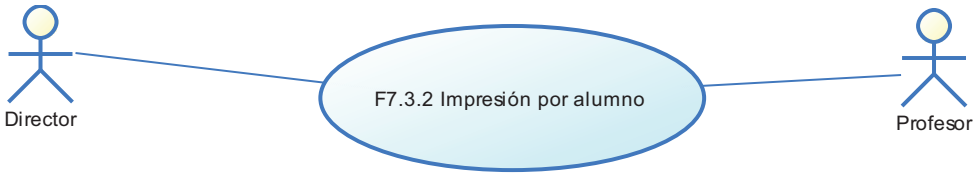
Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Error de impresión.	Intentar nuevamente.

- **F7.3.2 Impresión por alumno**

A través de este caso uso se imprime el carnet de un alumno específico.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir carnet por alumno.	**2**	El sistema despliega una ventana para el ingreso del código.	
3	El director o el profesor, ingresan el código del estudiante.			
4	El director o el profesor, dan clic en el botón Aceptar.	**5**	El sistema despliega el carnet del alumno.	**E1** **E2**
6	El director o el profesor, dan clic en el botón Imprimir.	**7**	El sistema envía a imprimir la información.	**E3**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Los datos ingresados son incorrectos.	Ingresar datos de nuevo.
E3	Error de impresión.	Intentar nuevamente.

Flujo alterno:

Paso	Descripción
5	Si no existe, verificar F2.1

- **F7.4 Imprimir los horarios**

  A través de este caso uso se imprime el horario de un curso.

- **Actores:** Director y profesor



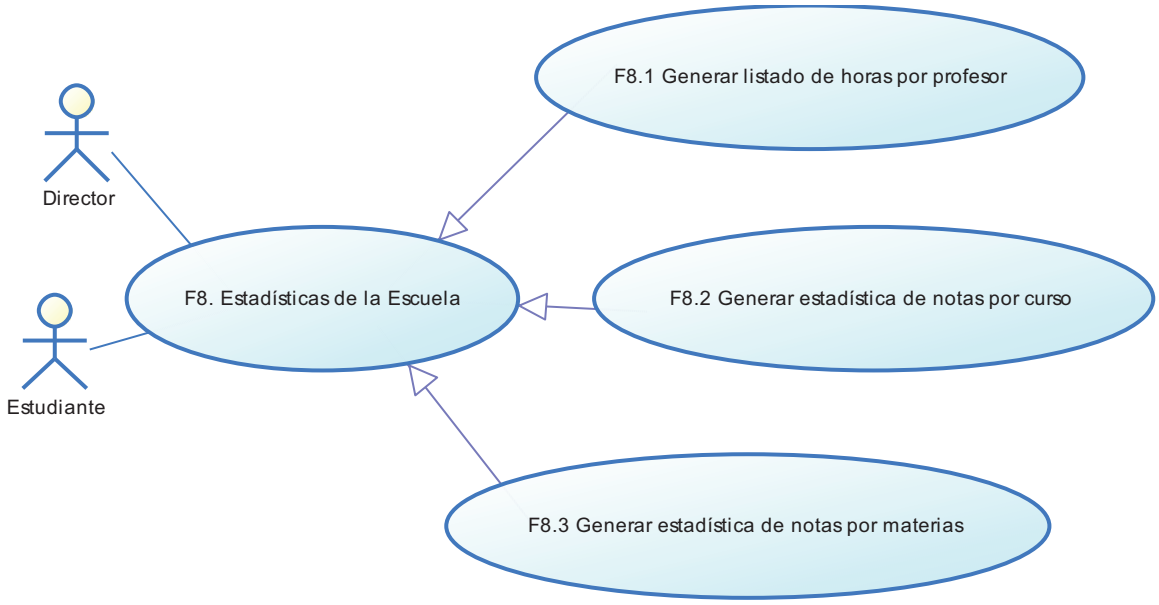
Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de imprimir horarios.	**2**	El sistema despliega un listado de cursos.	
3	El director o el profesor, escogen el curso de la lista.			
4	Presiona el botón Aceptar	**5**	El sistema muestra una ventana con el horario del curso elegido.	**E1**
6	El director o el profesor, dan clic en el botón Imprimir.	**7**	El sistema envía a imprimir la información.	**E2**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.
E2	Error de impresión.	Intentar nuevamente.

- **F8 Estadísticas de la Escuela**

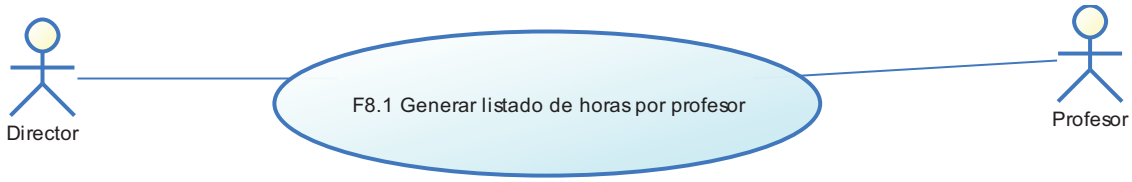


Detalle:

- **F8.1 Generar listado de horas por profesor**

  A través de este caso uso se genera un listado de horas por semana que dicta cada profesor.

Actores: Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de Generar listado de horas por profesor.	**2**	El sistema despliega un listado de profesores.	**E1**

3	El director o el profesor, escogen al profesor que desea la estadística de la lista.			
4	Presiona botón Aceptar	**5**	El sistema muestra una ventana con los datos solicitados.	**E1**

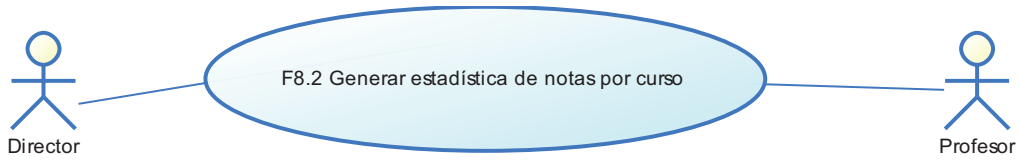
Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F8.2 Generar estadística de notas por curso**

  A través de este caso uso se genera un listado los promedios de notas de todos los cursos.

- **Actores:** Director y profesor



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de Generar estadística de notas por curso.	**2**	El sistema despliega una ventana de los cursos con sus promedios.	**E1**

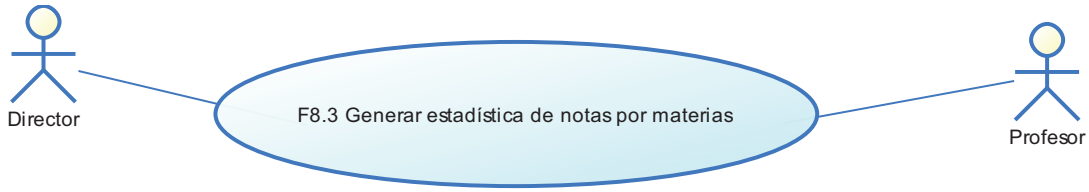
Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F8.3 Generar estadística de notas por materias**

  A través de este caso uso se genera un listado de las materias con sus respectivas notas, de un curso específico.

Actores: Director y profesor



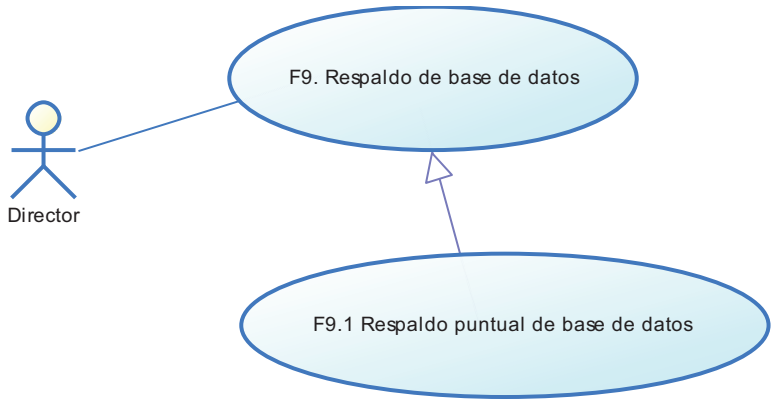
Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director o el profesor, escogen la opción de Generar estadística de notas por materia.	**2**	El sistema despliega una lista de los cursos.	
3	El director o el profesor, escogen un curso de la lista.			
4	Presiona botón Aceptar	**5**	El sistema despliega la estadística de las notas por cursos	**E1**

Excepciones:

Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

- **F9. Respaldo de base de datos**

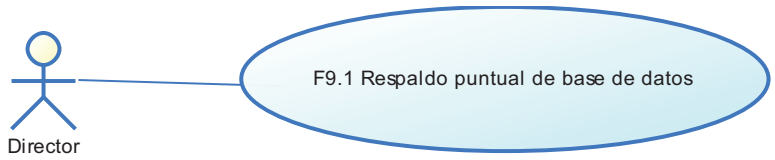


Detalle:

- **F9.1 Respaldo puntual periódico de base de datos**

  A través de este caso uso se realiza un respaldo de base de datos cuando se crea necesario.

- **Actores:** Director



Descripción:

Paso	Actor	Paso	Sistema	Exep.
1	El director escoge la opción de Respaldo periódico de base de datos	**2**	El sistema pregunta el destino donde se va a guardar el respaldo.	
3	El director escoge el lugar donde se va a guardar.			
4	El director da clic en el	**5**	El sistema respalda la base de datos	**E1**

botón Respaldar.	en el destino.

Excepciones:

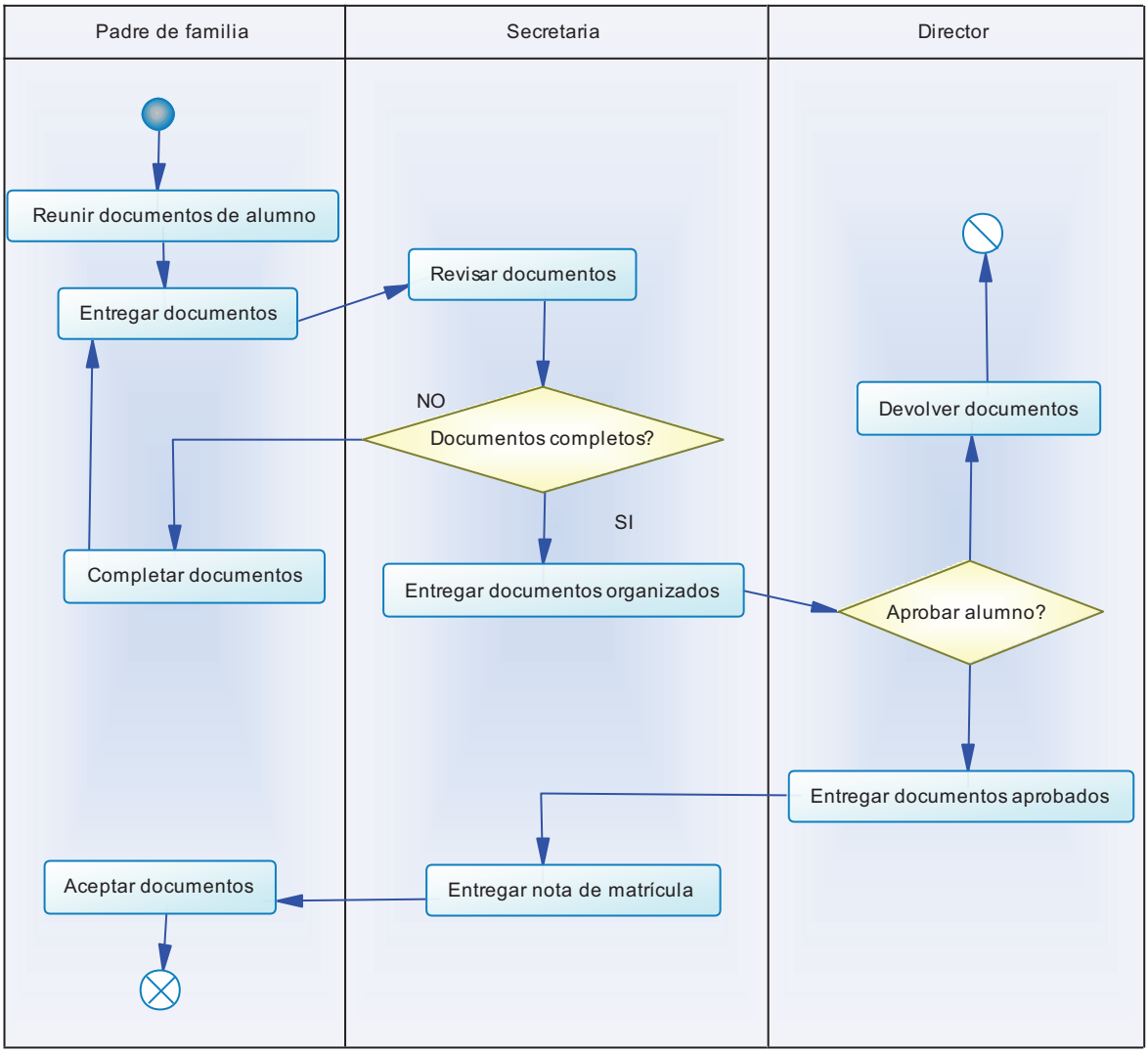
Código	Descripción	Alternativa
E1	No hay conexión a la base de datos.	Intentar nuevamente.

3.2 DISEÑO DE PROCESOS

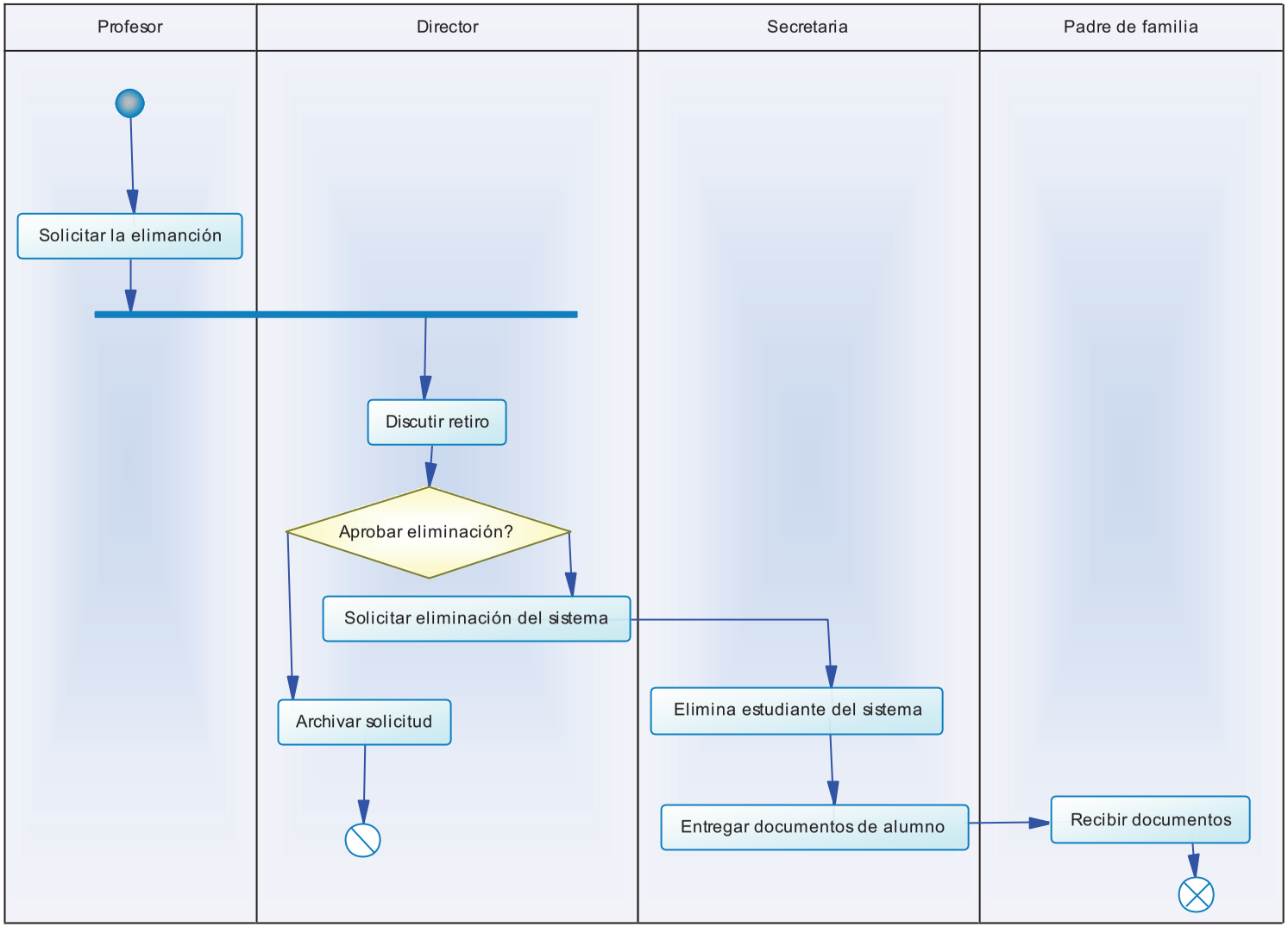
3.2.1 Diagramación de Procesos

Se van a utilizar diagramas de activadas para demostrar paso a paso como van a ir los flujos de las funciones que va a realizar el programa.

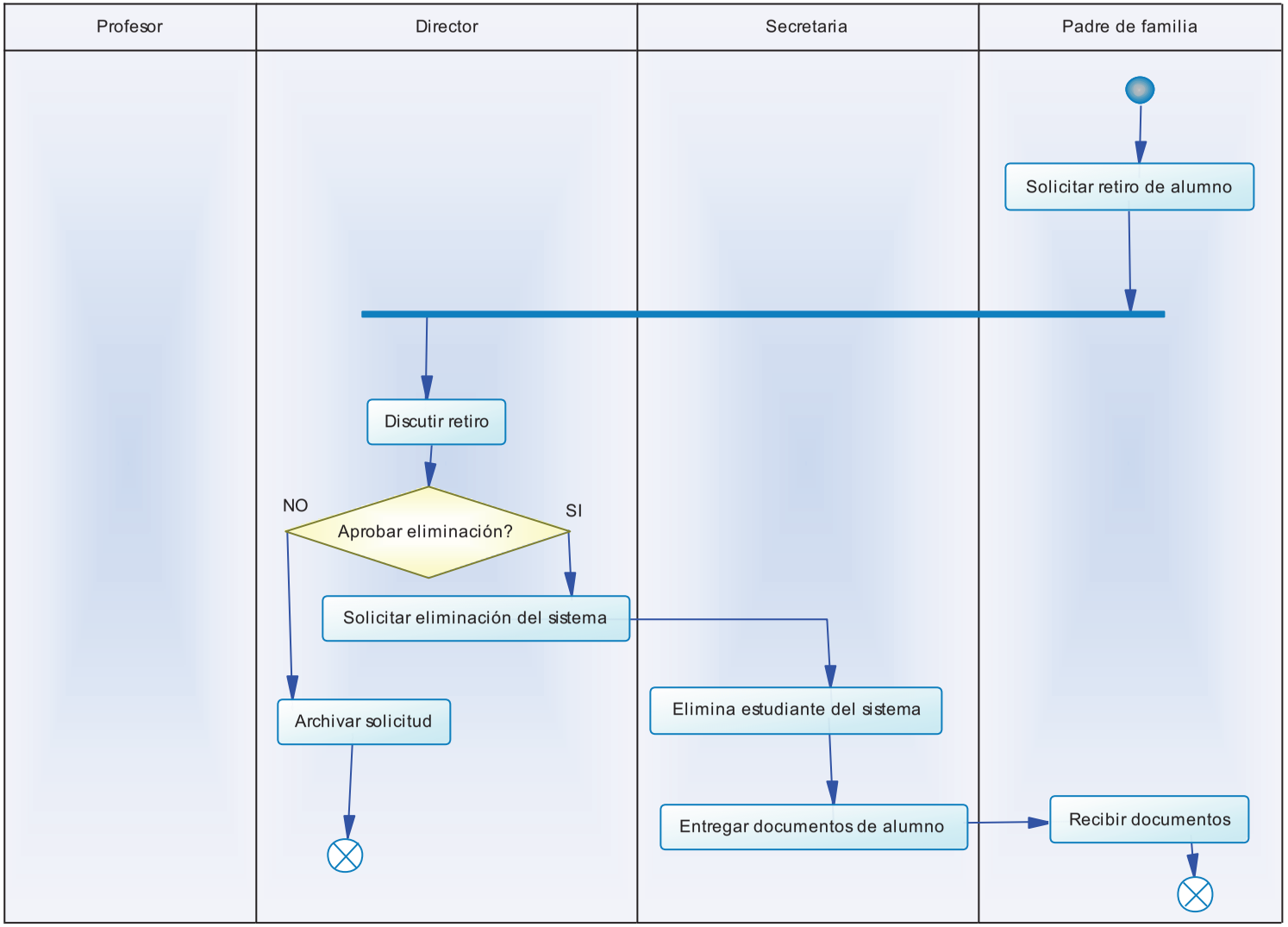
- **Proceso de matriculación (F2.1 Ingreso de estudiantes)**



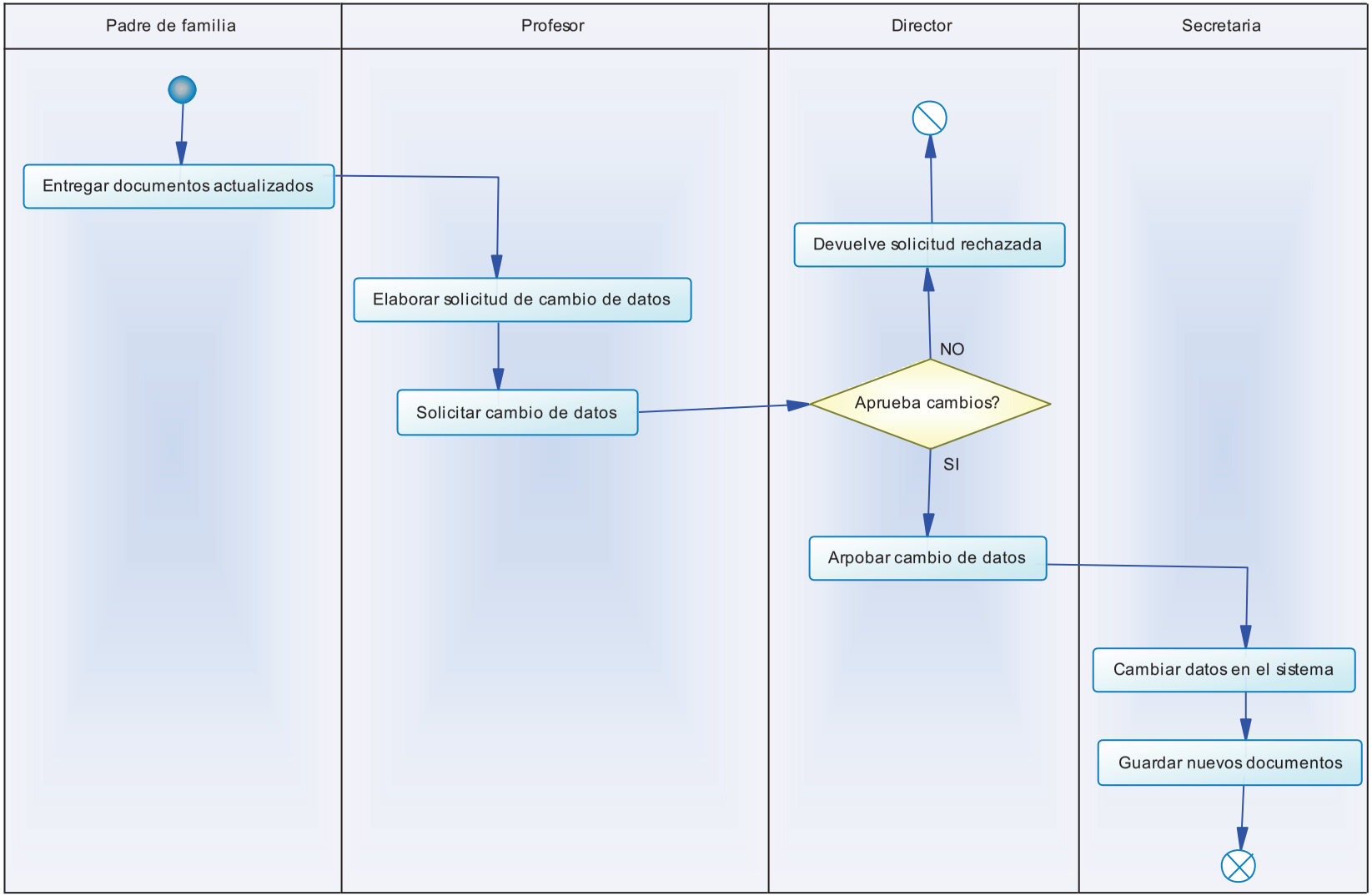
- **Proceso de salida de estudiante (F1.2 Eliminar estudiante) (Profesor)**



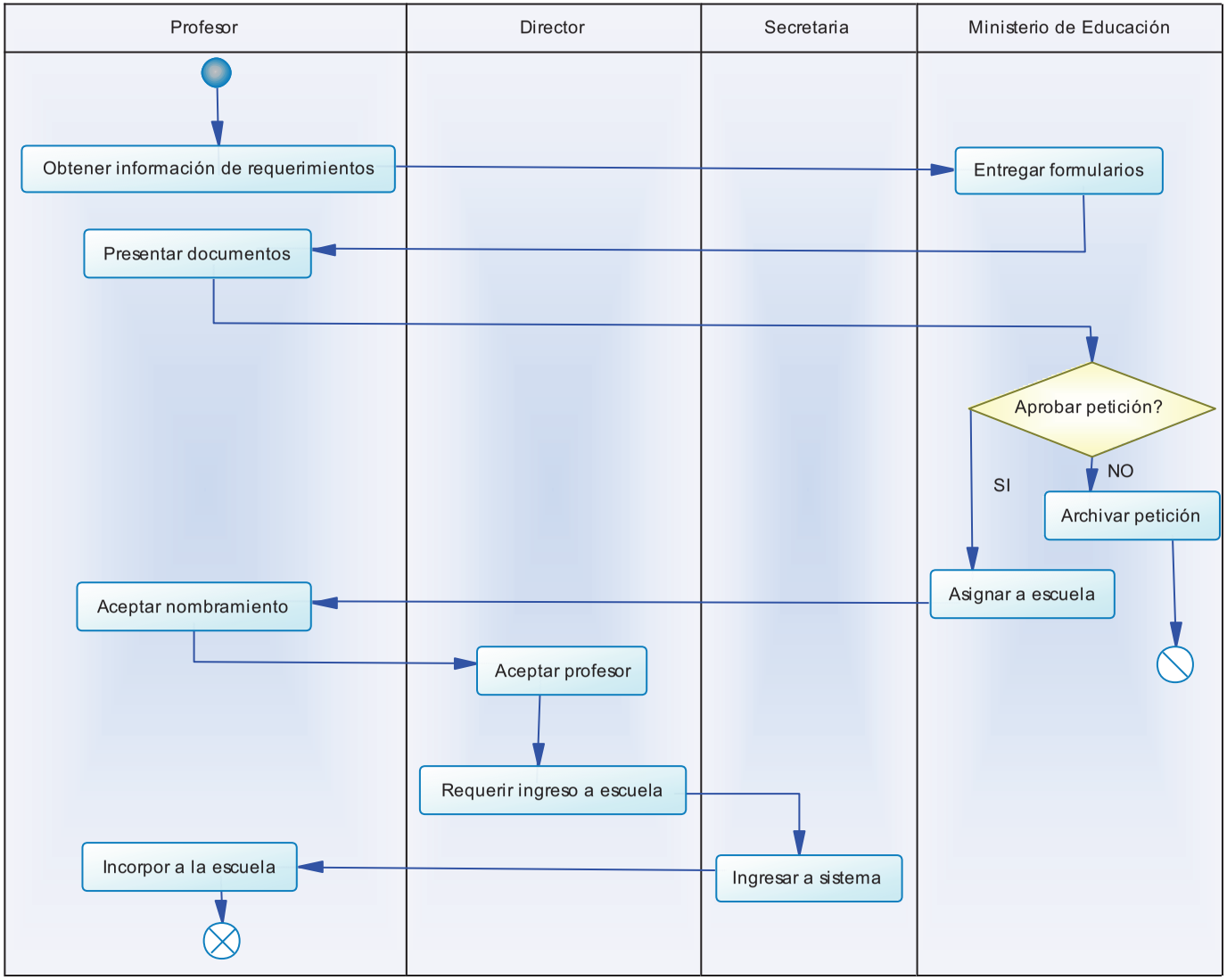
- **Proceso de salida de estudiante (F1.2 Eliminar estudiante) (Padre de familia)**



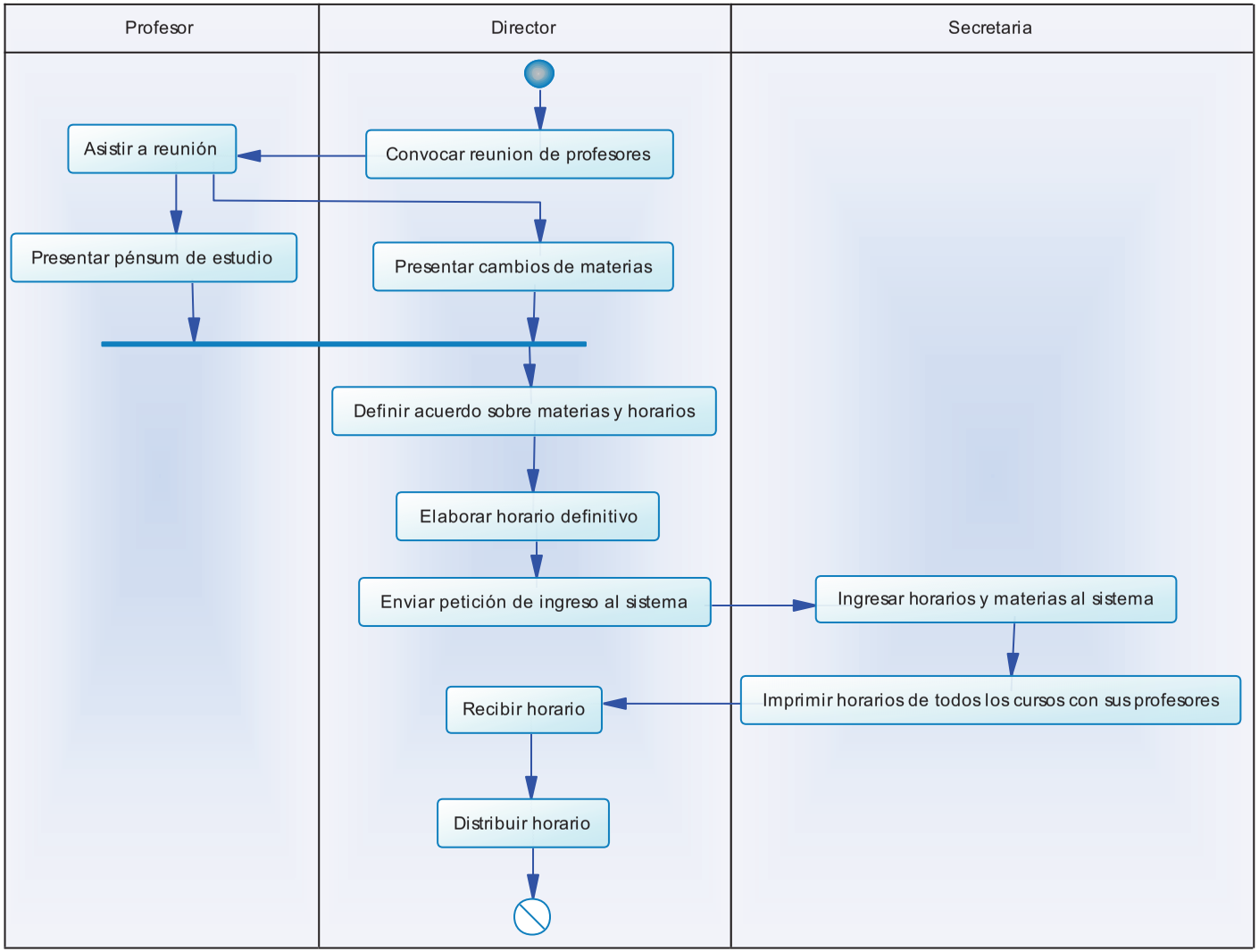
- **Proceso de actualización de datos (F2.3 Actualizar estudiantes)**



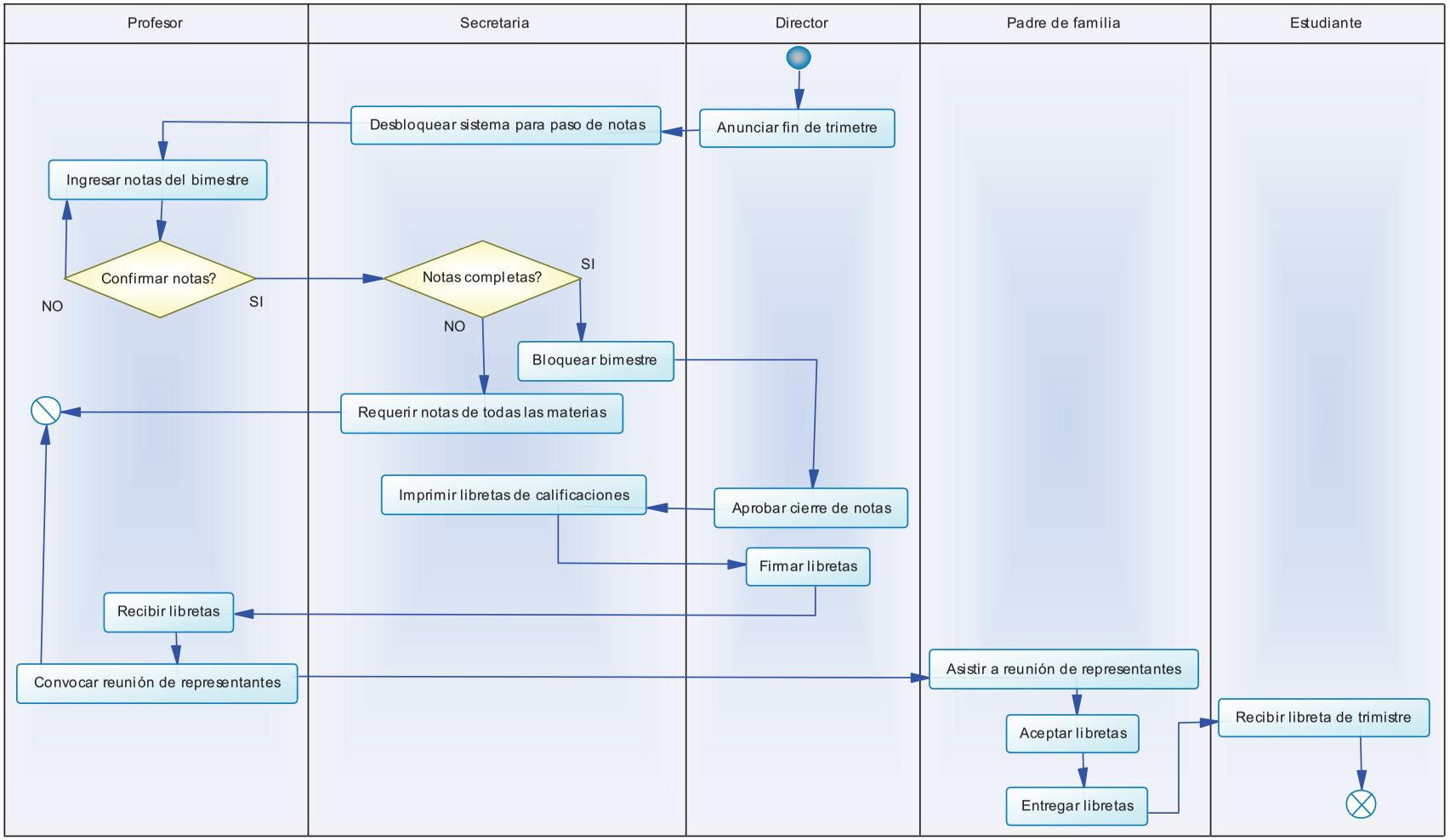
- **Proceso contratación de profesores (F3.1 Ingresar profesor)**



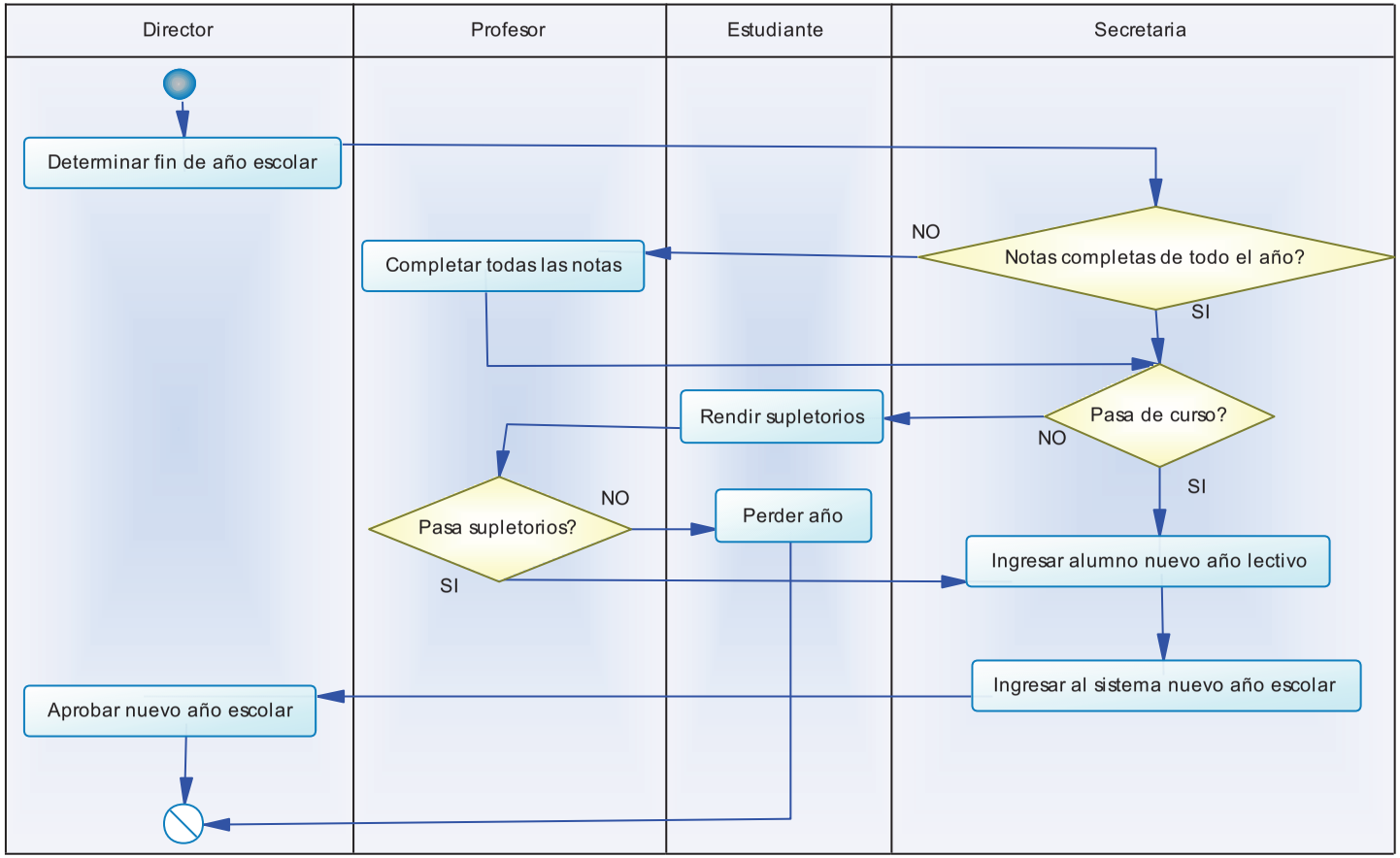
- **Proceso creación de horarios escolares (F5.1 Ingresar materia)**



- **Proceso creación de libretas escolares (F6.1 Generar libreta de calificaciones)**

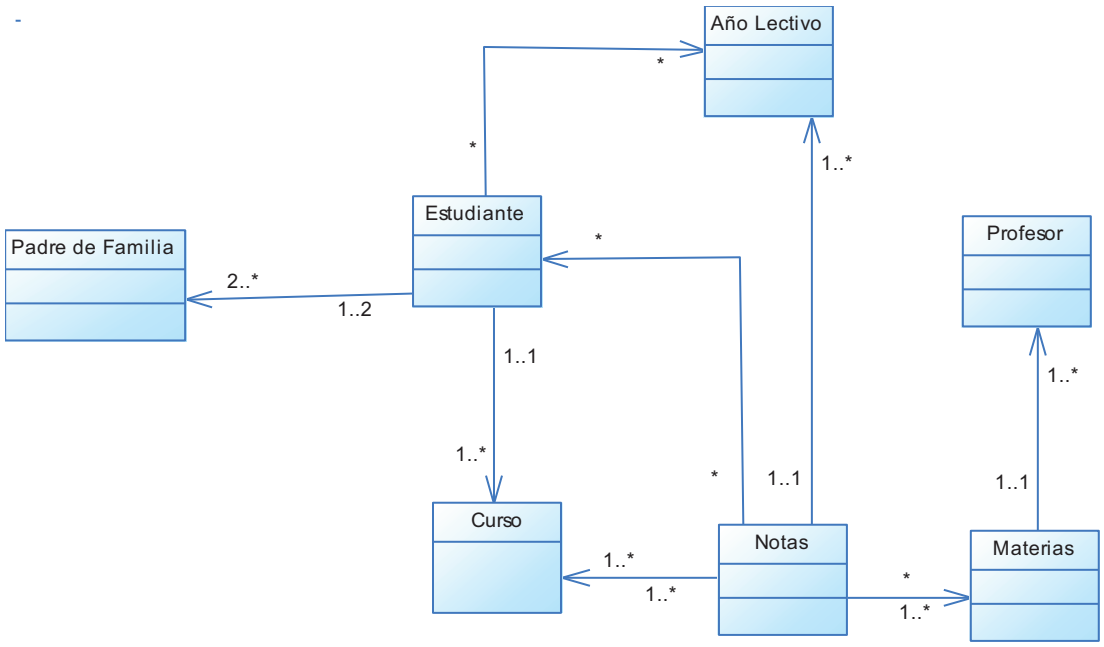


- **Proceso cambio a nuevo año escolar (F6.2 Generar nuevo año escolar)**

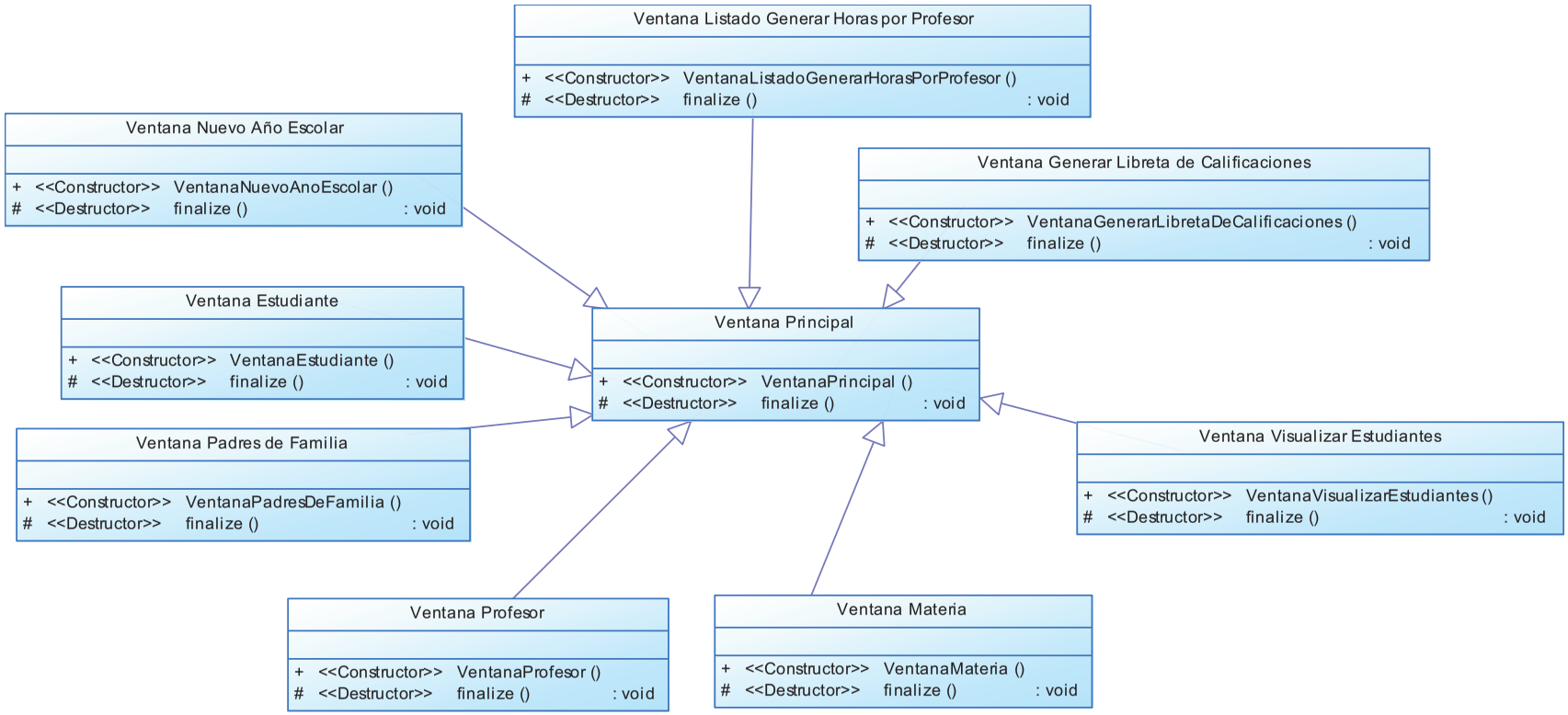


3.2.2 Diagramación de Clases

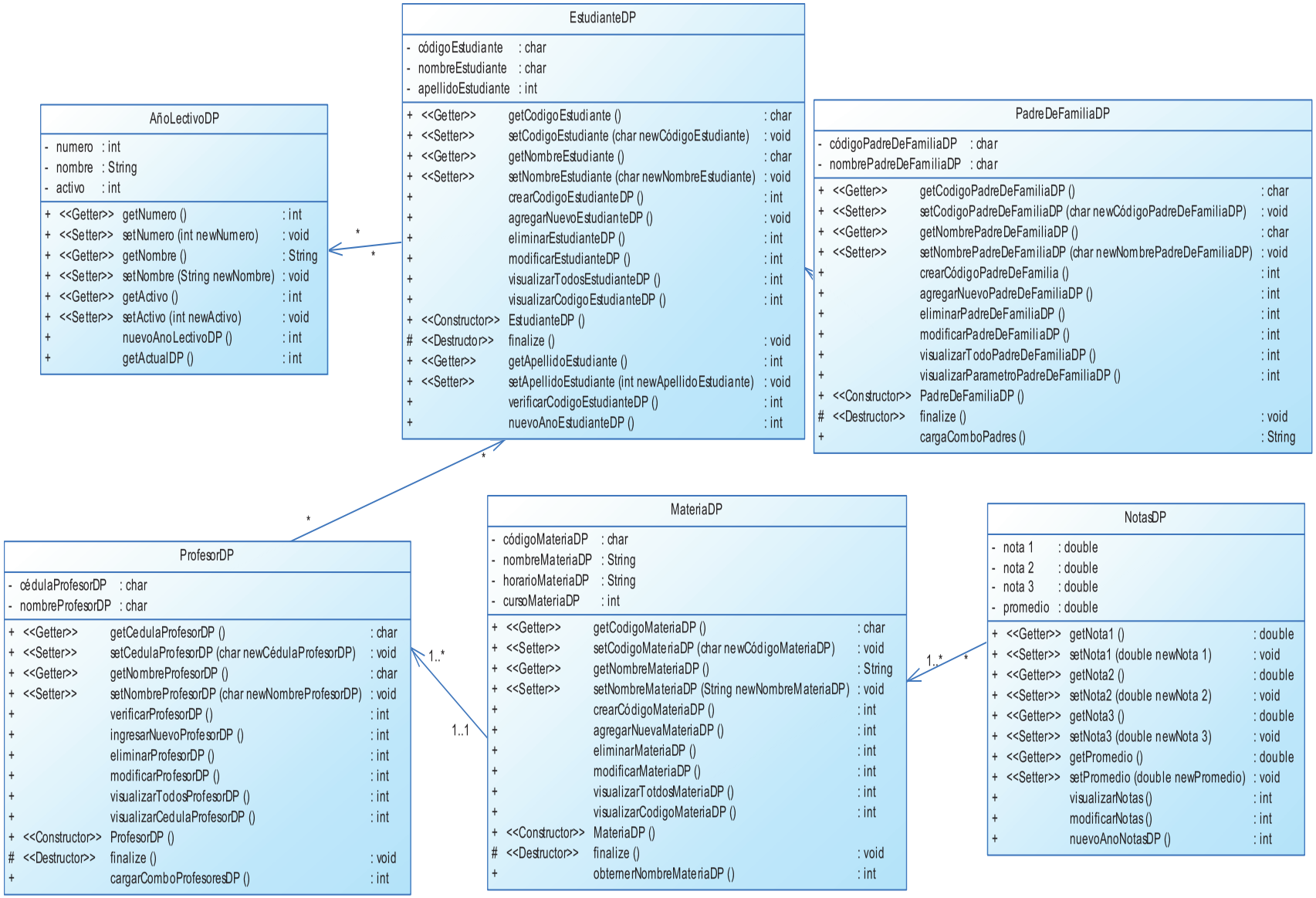
Se presenta el diagrama de clases, para mostrar la estructura del sistema basado en sus clases y las asociaciones entre ellos.



- **Diagrama de GUI (interfaz de usuario)**



- **Diagrama de dominio del problema (DP)**

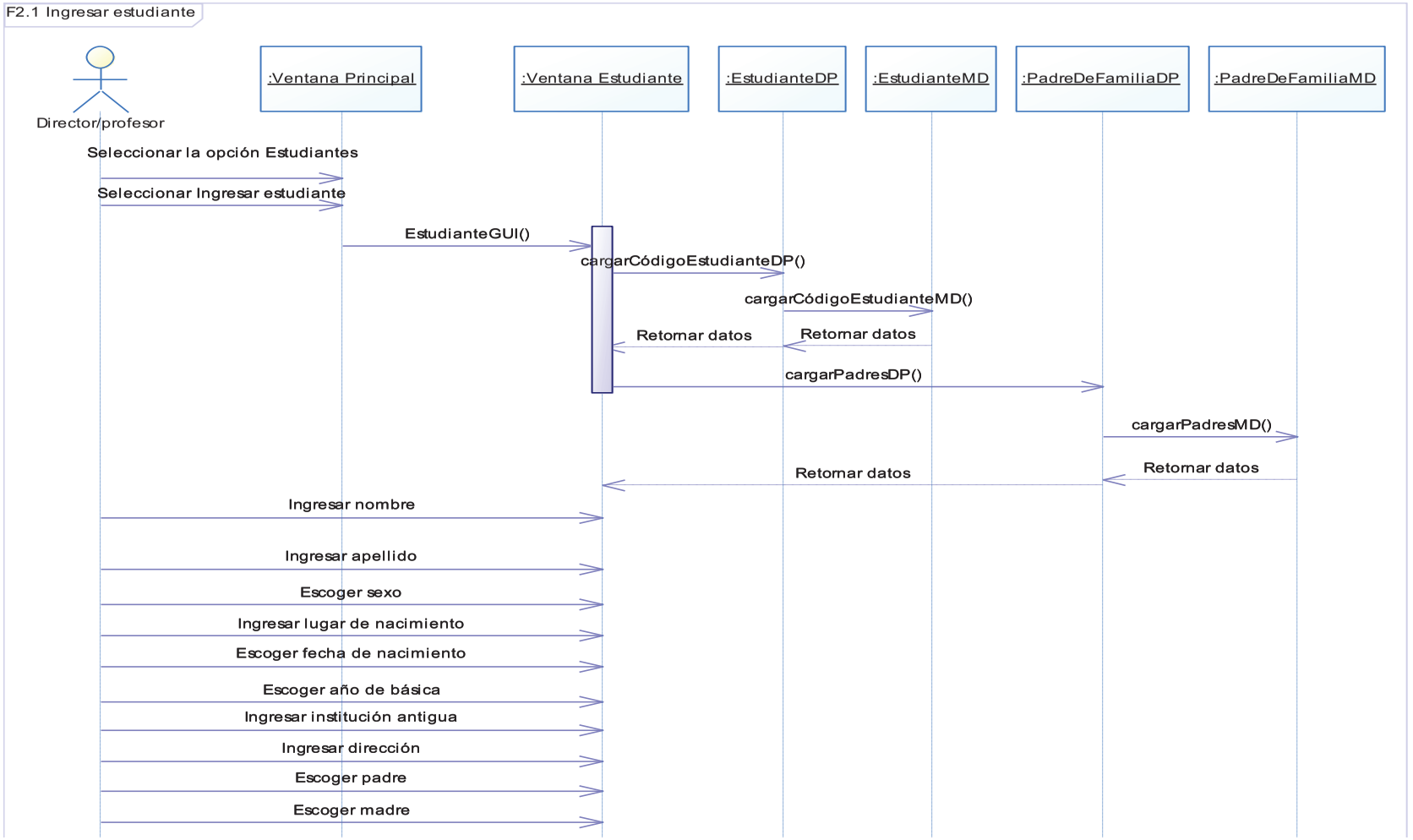


- **Diagrama de manejo de datos (MD)**

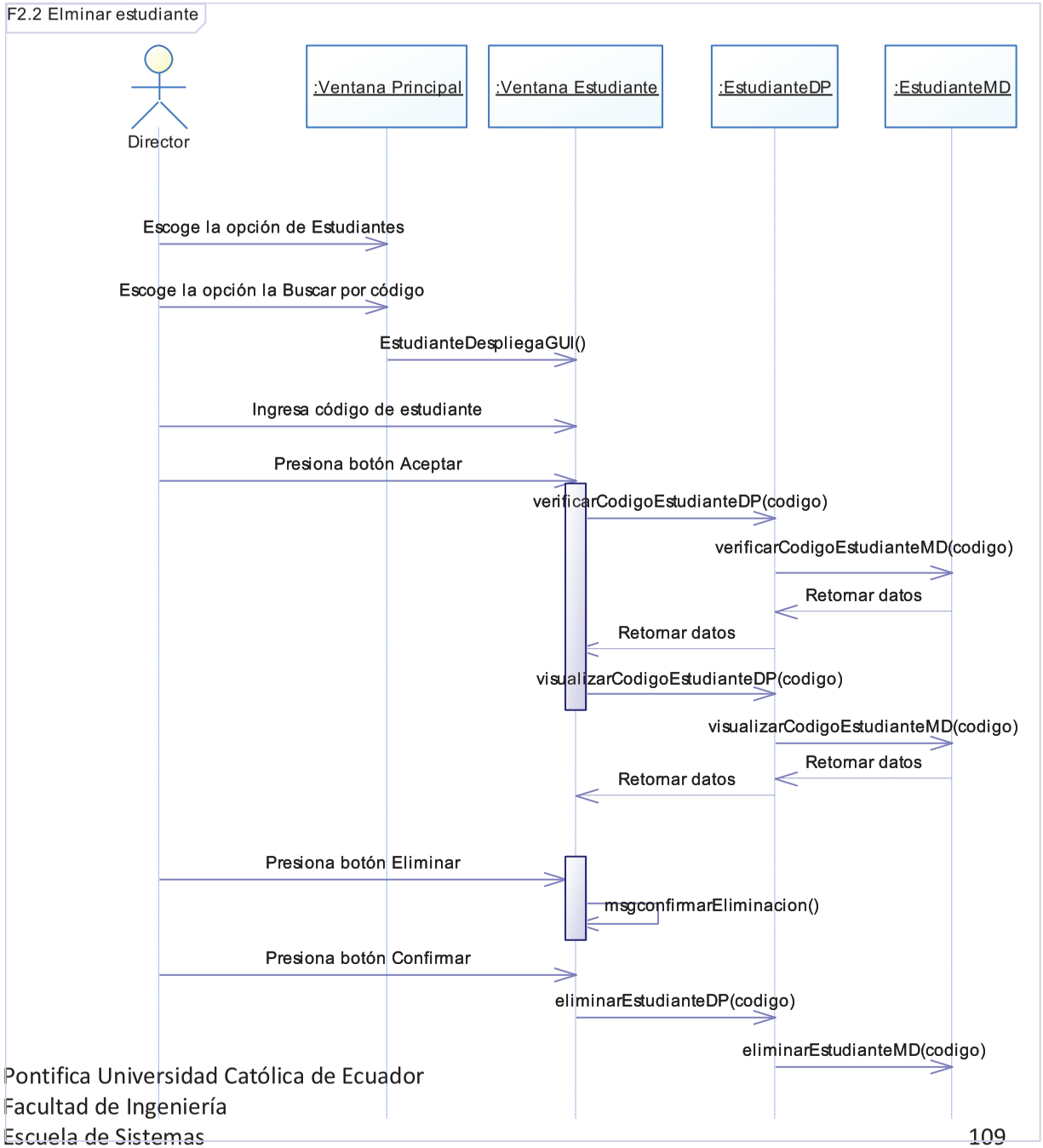


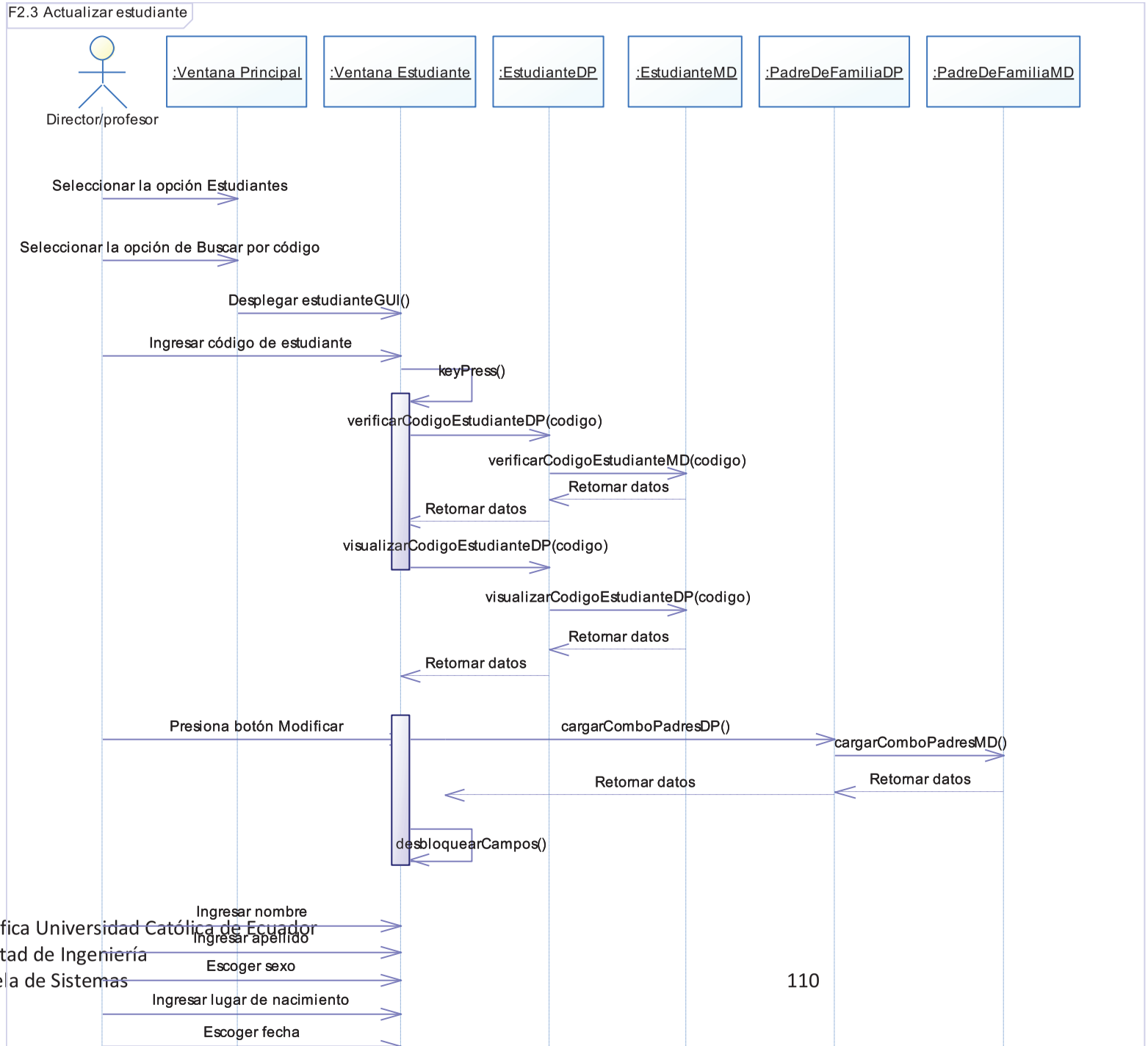
3.2.3 Diagramas de secuencia

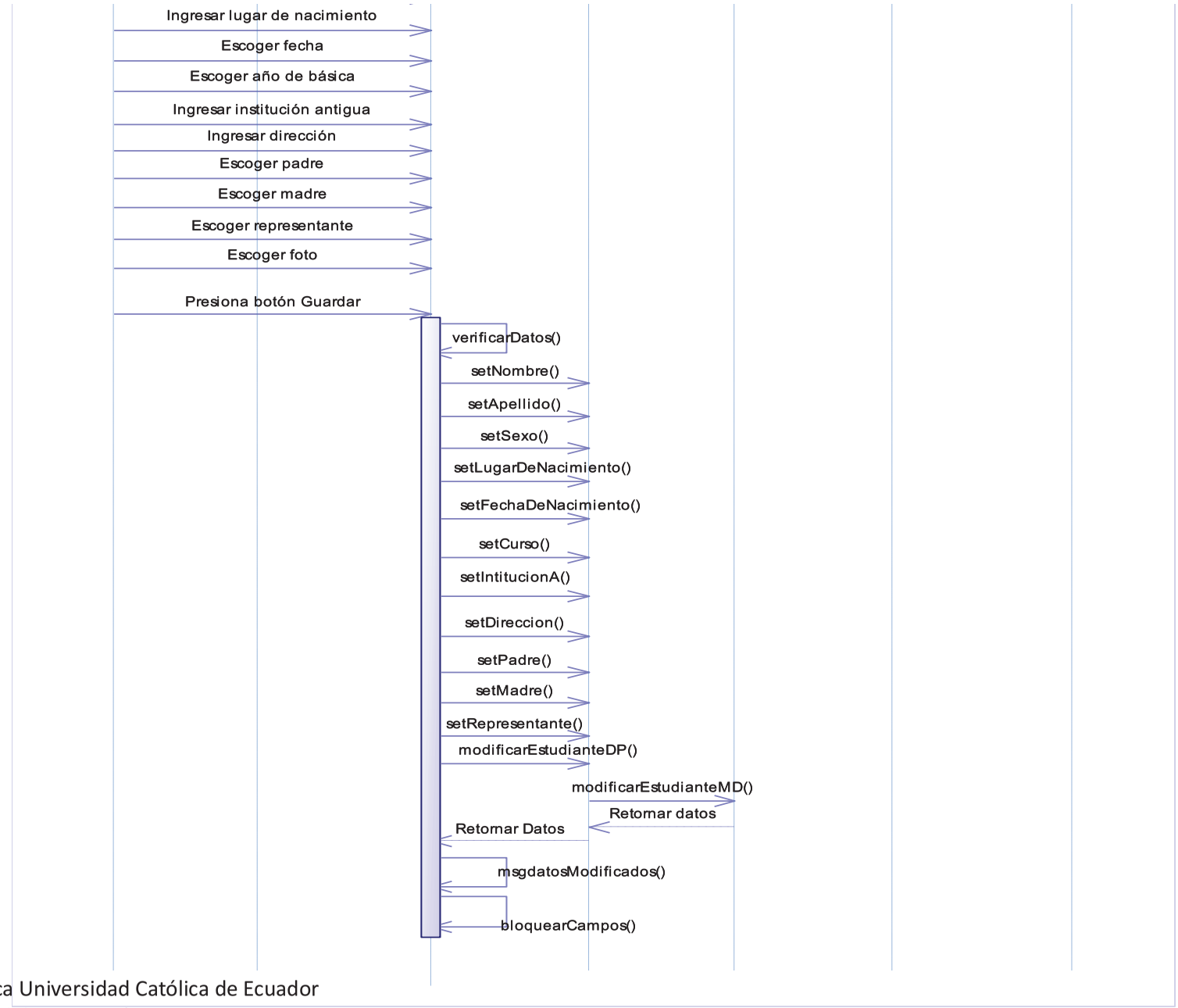
Los diagramas de secuencia nos permite definir la interacción de todos los objetos que componen la aplicación según van siendo llamados o requeridos, para tener una idea puntual de cada caso.

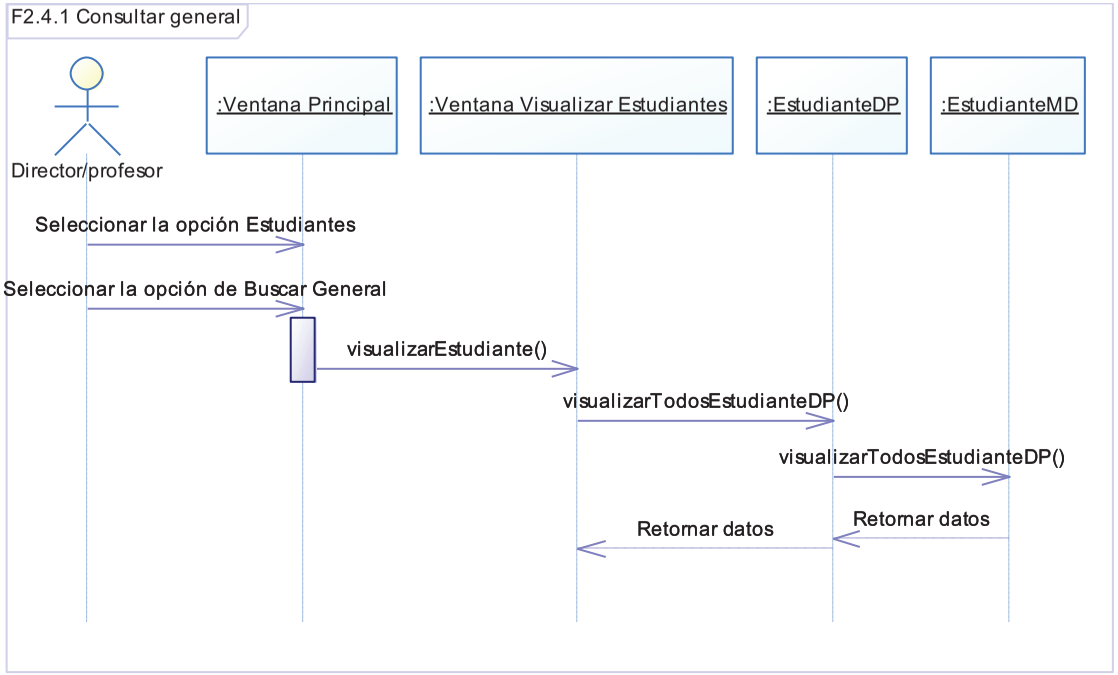


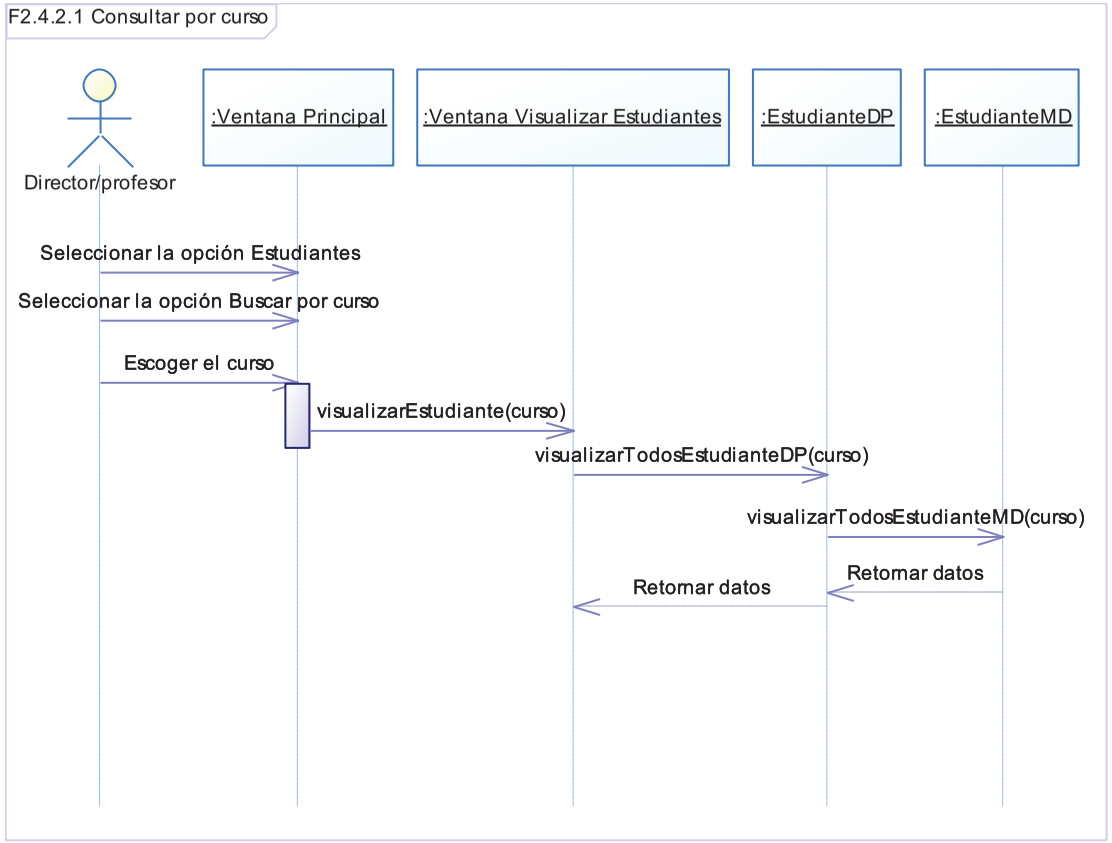

Escoger representante
Escoger foto
Presiona botón Guardar
verificarDatos
setNombre()
setSexo()
setApellido()
setLugarDeNacimiento()
setFechaDeNacimiento()
setInstitucionA()
setPadre()
setMadre()
setRepresentante
setFoto()
agregarNuevoEstudianteMD(codigo,nombre,apellido)

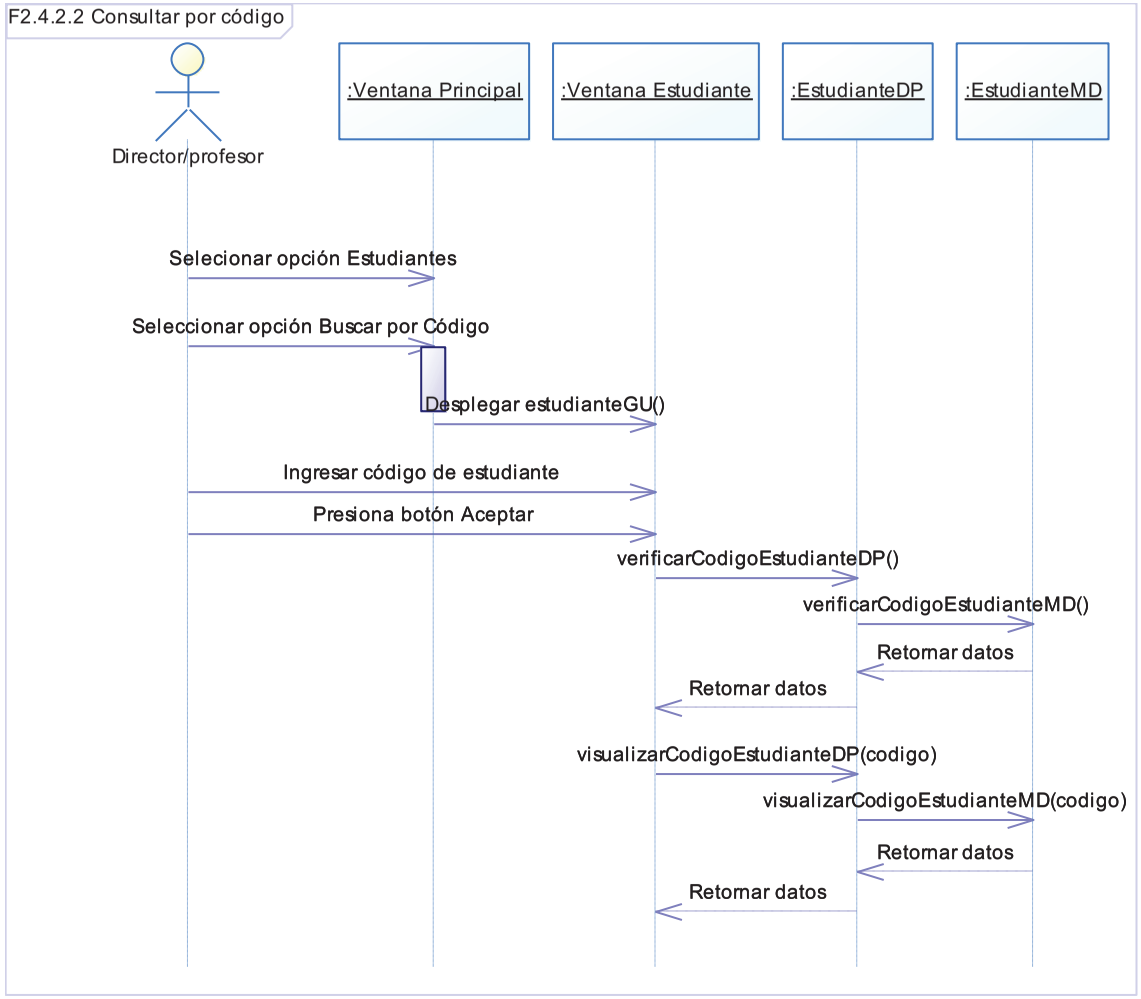

F2.2 Elminar estudiante
Director
:Ventana Principal
:Ventana Estudiante
:EstudianteDP
:EstudianteMD
Escoge la opción de Estudiantes
Escoge la opción la Buscar por código
EstudianteDespliegaGUI()
Ingresa código de estudiante
Presiona botón Aceptar
verificarCodigoEstudianteDP(codigo)
verificarCodigoEstudianteMD(codigo)
Retornar datos
Retornar datos
visualizarCodigoEstudianteDP(codigo)
visualizarCodigoEstudianteMD(codigo)
Retornar datos
Retornar datos
Presiona botón Eliminar
msgconfirmarEliminacion()
Presiona botón Confirmar
eliminarEstudianteDP(codigo)
eliminarEstudianteMD(codigo)

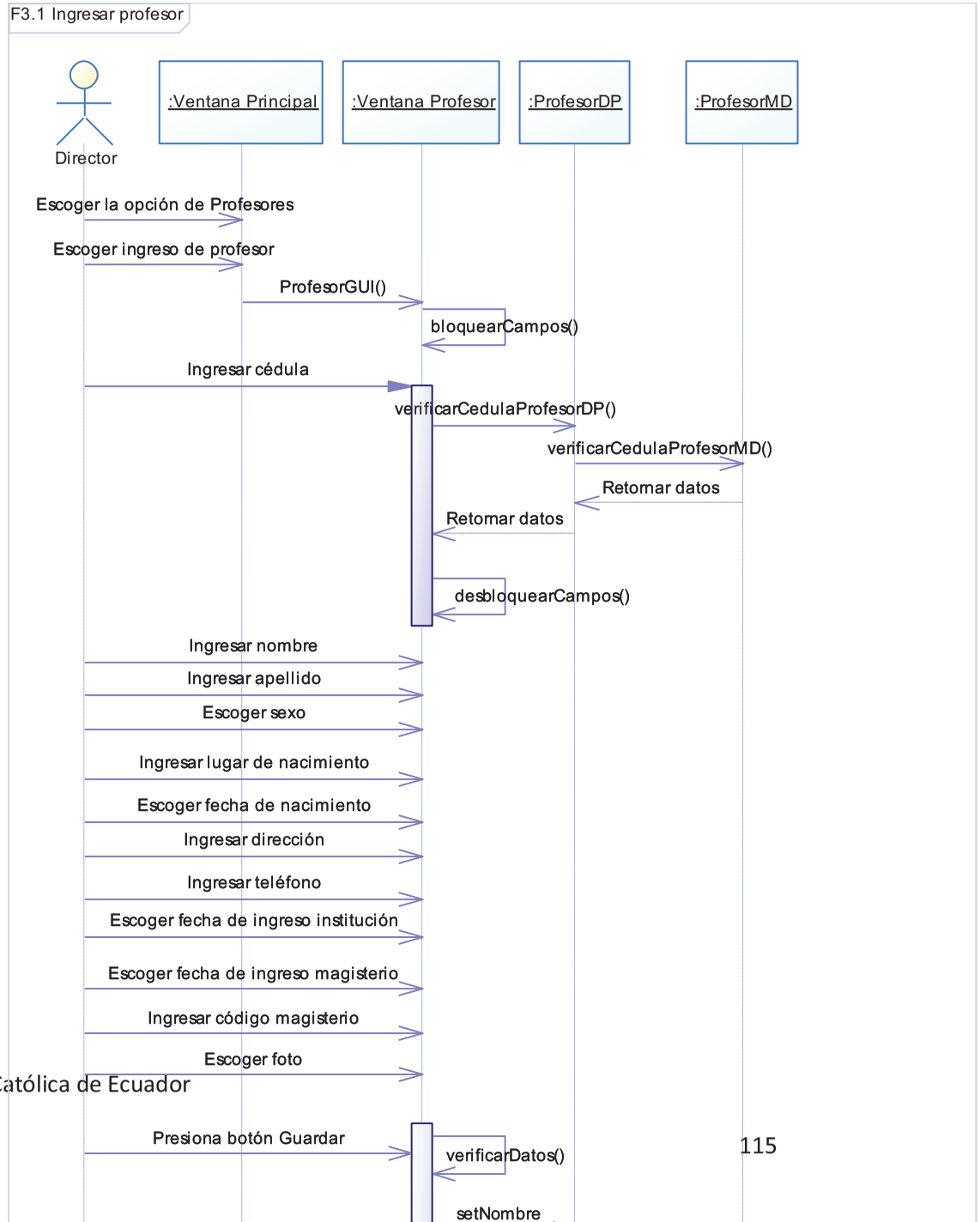

F2.3 Actualizar estudiante
Director/profesor
:Ventana Principal
:Ventana Estudiante
:EstudianteDP
:EstudianteMD
:PadreDeFamiliaDP
:PadreDeFamiliaMD
Seleccionar la opción Estudiantes
Seleccionar la opción de Buscar por código
Desplegar estudianteGUI()
Ingresar código de estudiante
keyPress()
verificarCodigoEstudianteDP(codigo)
verificarCodigoEstudianteMD(codigo)
Retornar datos
Retornar datos
visualizarCodigoEstudianteDP(codigo)
visualizarCodigoEstudianteDP(codigo)
Retornar datos
Retornar datos
Presiona botón Modificar
cargarComboPadresDP()
cargarComboPadresMD()
Retornar datos
Retornar datos
desbloquearCampos()
Ingresar nombre
Ingresar apellido
Escoger sexo
Ingresar lugar de nacimiento
Escoger fecha

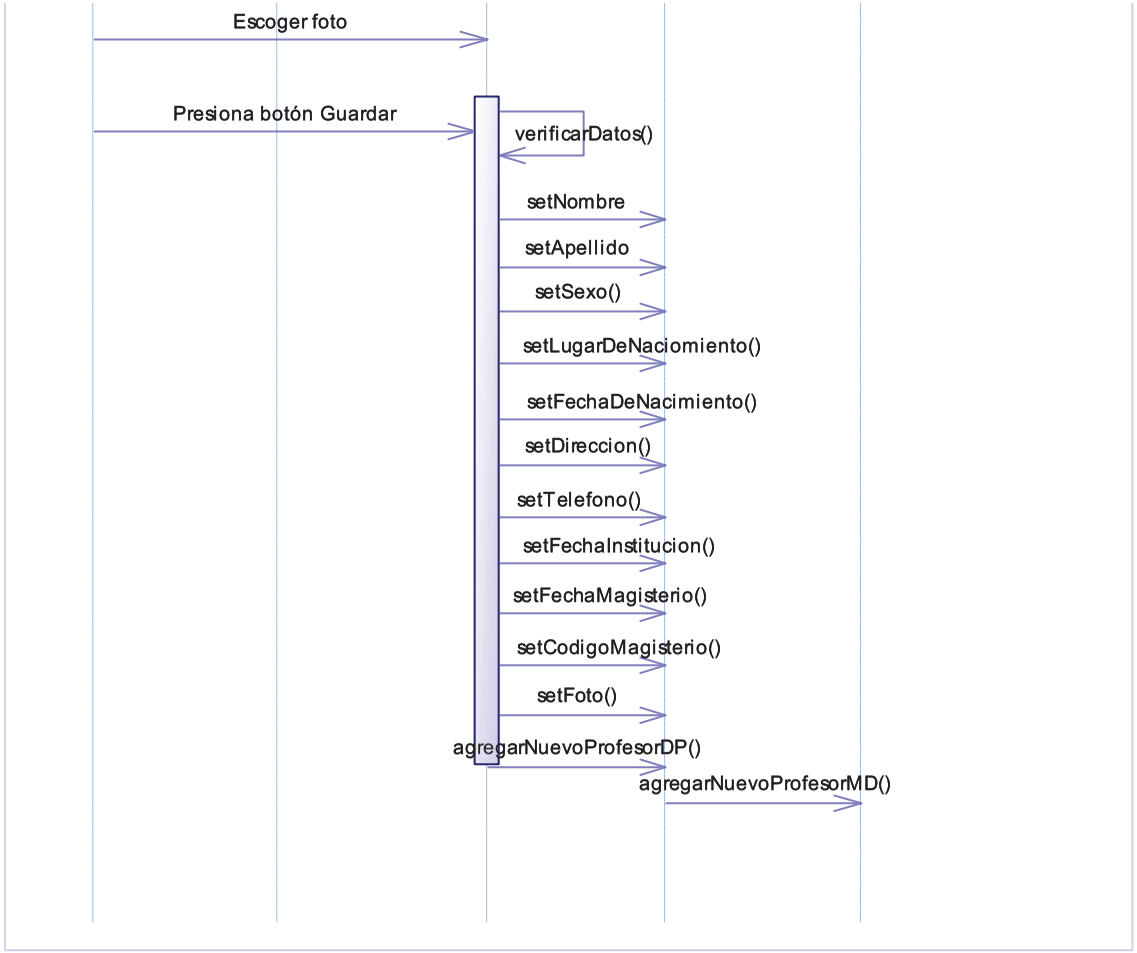

Ingresar lugar de nacimiento
Escoger fecha
Escoger año de básica
Ingresar institución antigua
Ingresar dirección
Escoger padre
Escoger madre
Escoger representante
Escoger foto
Presiona botón Guardar
verificarDatos()
setNombre()
setApellido()
setSexo()
setLugarDeNacimiento()
setFechaDeNacimiento()
setCurso()
setIntitucionA()
setDireccion()
setPadre()
setMadre()
setRepresentante()
modificarEstudianteDP()
modificarEstudianteMD()
Retornar datos
Retornar Datos
msgdatosModificados()
bloquearCampos()

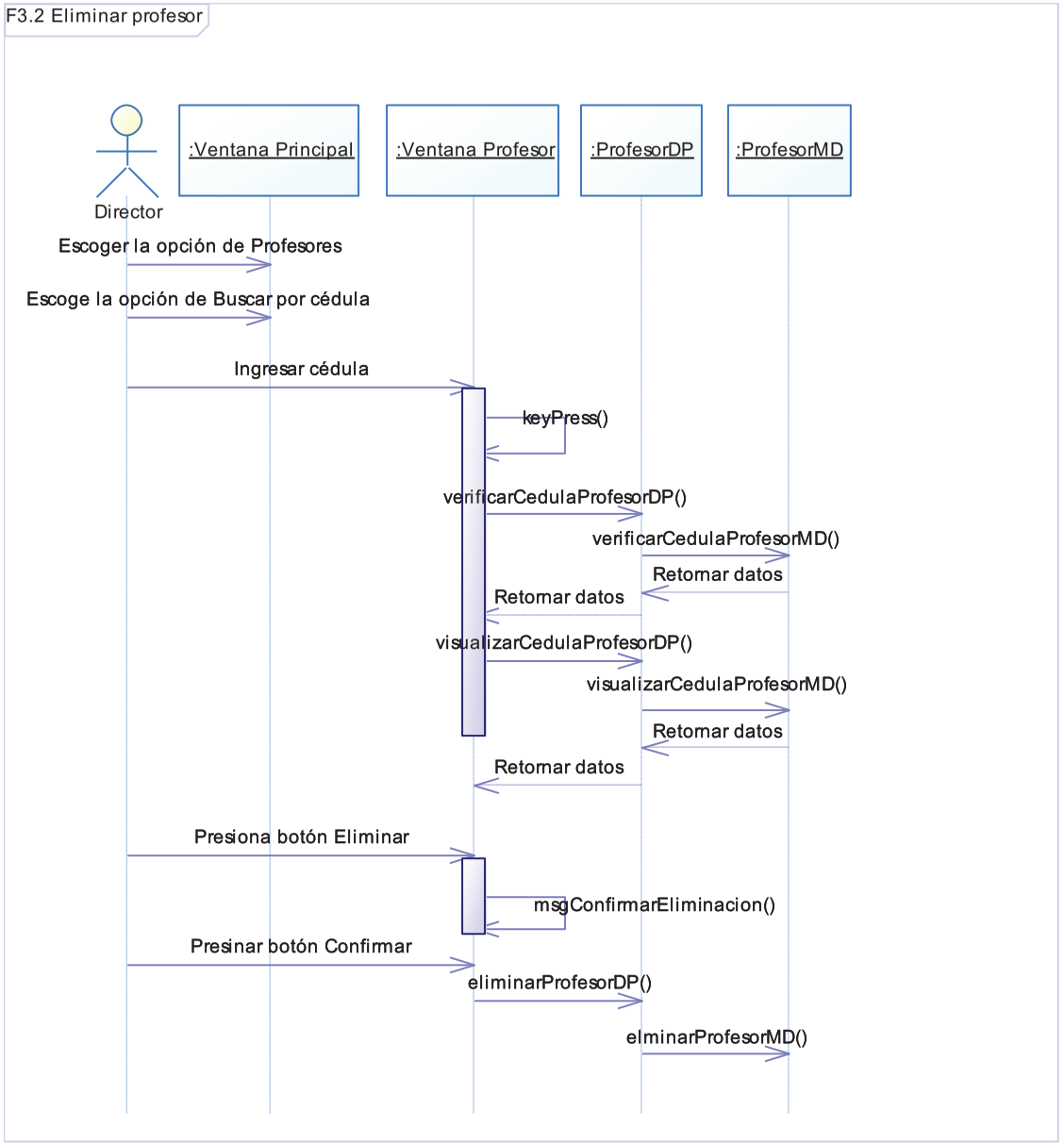

F2.4.1 Consultar general
Director/profesor
:Ventana Principal
:Ventana Visualizar Estudiantes
:EstudianteDP
:EstudianteMD
Seleccionar la opción Estudiantes
Seleccionar la opción de Buscar General
visualizarEstudiante()
visualizarTodosEstudianteDP()
visualizarTodosEstudianteDP()
Retornar datos
Retornar datos

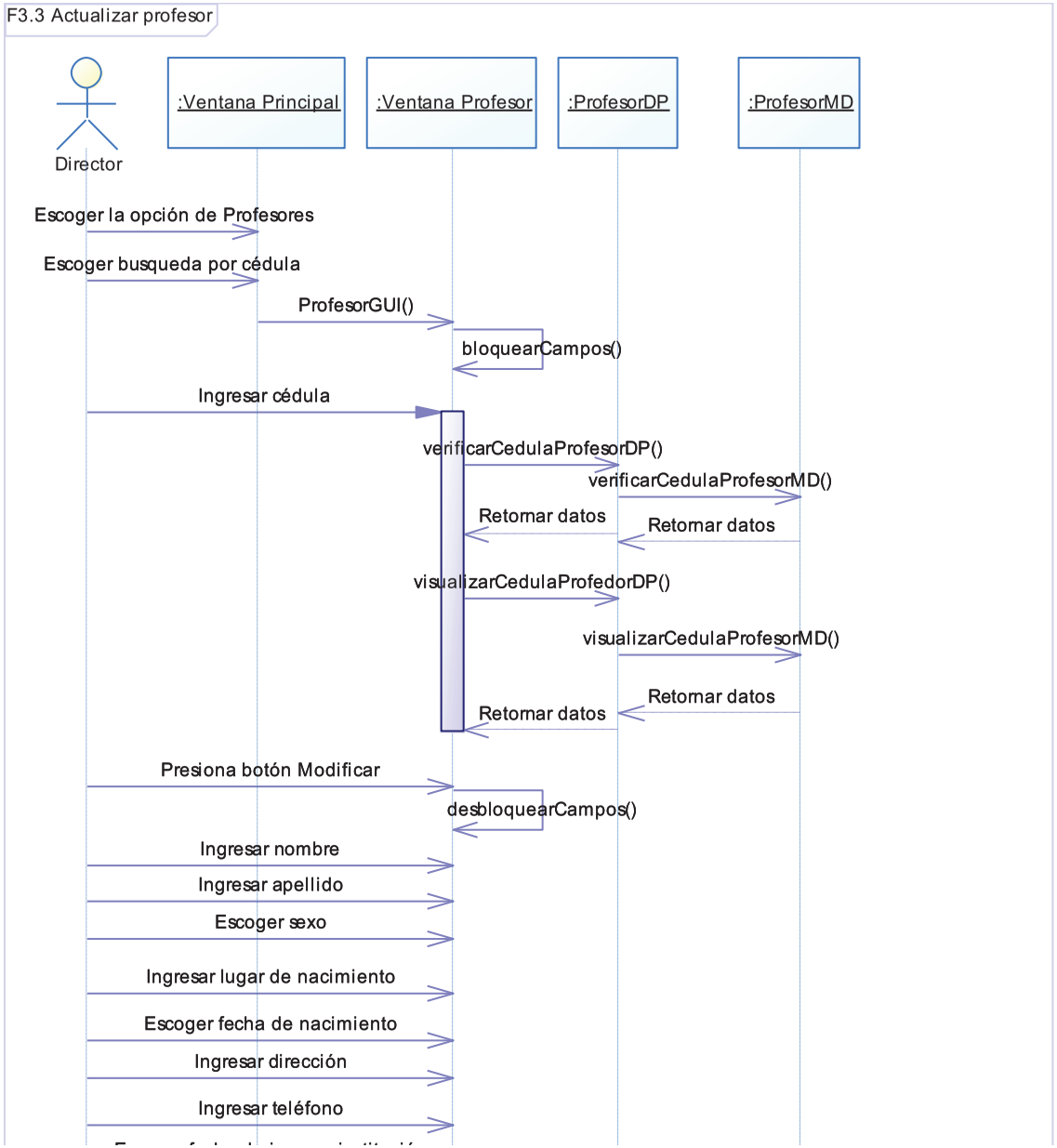

F2.4.2.1 Consultar por curso
Director/profesor
:Ventana Principal
:Ventana Visualizar Estudiantes
:EstudianteDP
:EstudianteMD
Seleccionar la opción Estudiantes
Seleccionar la opción Buscar por curso
Escoger el curso
visualizarEstudiante(curso)
visualizarTodosEstudianteDP(curso)
visualizarTodosEstudianteMD(curso)
Retornar datos
Retornar datos

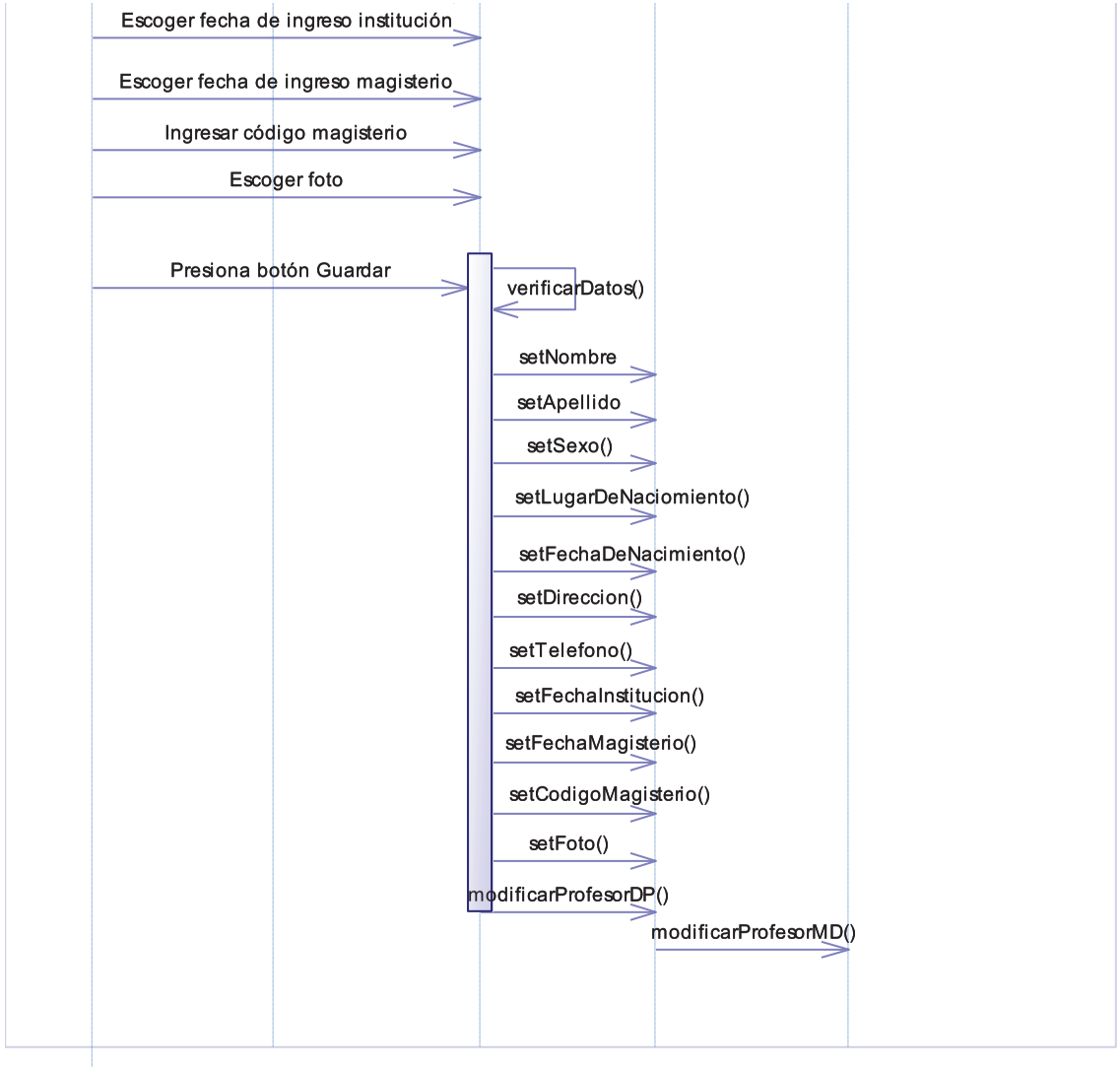

F2.4.2.2 Consultar por código
Director/profesor
:Ventana Principal
:Ventana Estudiante
:EstudianteDP
:EstudianteMD
Selecionar opción Estudiantes
Seleccionar opción Buscar por Código
Desplegar estudianteGU()
Ingresar código de estudiante
Presiona botón Aceptar
verificarCodigoEstudianteDP()
verificarCodigoEstudianteMD()
Retornar datos
Retornar datos
visualizarCodigoEstudianteDP(codigo)
visualizarCodigoEstudianteMD(codigo)
Retornar datos
Retornar datos

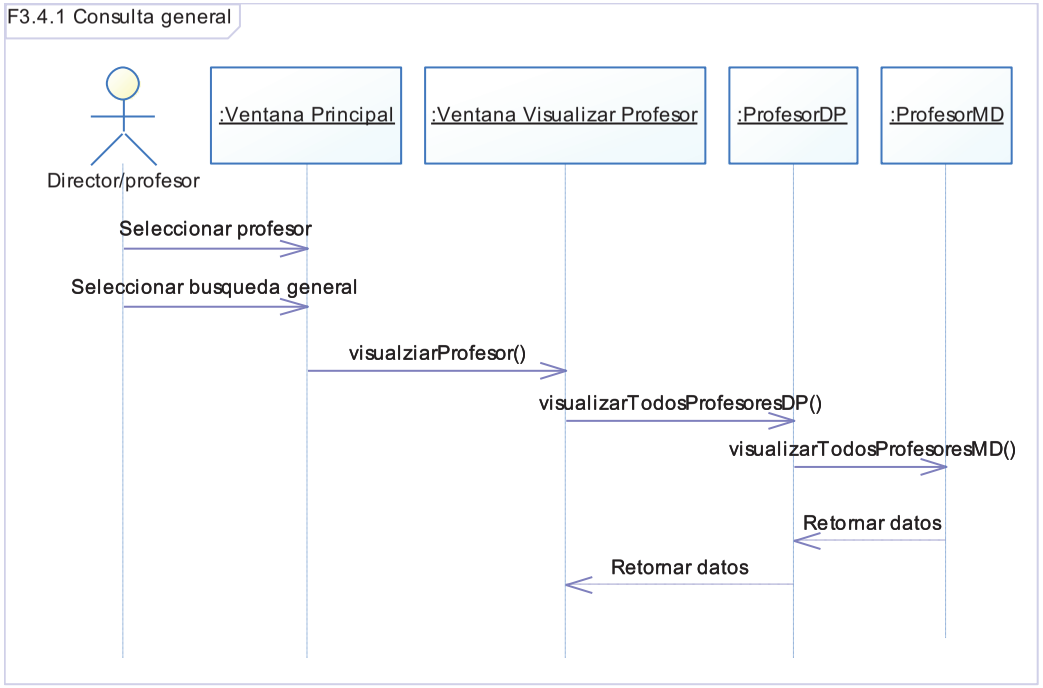

F3.1 Ingresar profesor
Director
:Ventana Principal
:Ventana Profesor
:ProfesorDP
:ProfesorMD
Escoger la opción de Profesores
Escoger ingreso de profesor
ProfesorGUI()
bloquearCampos()
Ingresar cédula
verificarCedulaProfesorDP()
verificarCedulaProfesorMD()
Retornar datos
Retornar datos
desbloquearCampos()
Ingresar nombre
Ingresar apellido
Escoger sexo
Ingresar lugar de nacimiento
Escoger fecha de nacimiento
Ingresar dirección
Ingresar teléfono
Escoger fecha de ingreso institución
Escoger fecha de ingreso magisterio
Ingresar código magisterio
Escoger foto
Presiona botón Guardar
verificarDatos()
setNombre

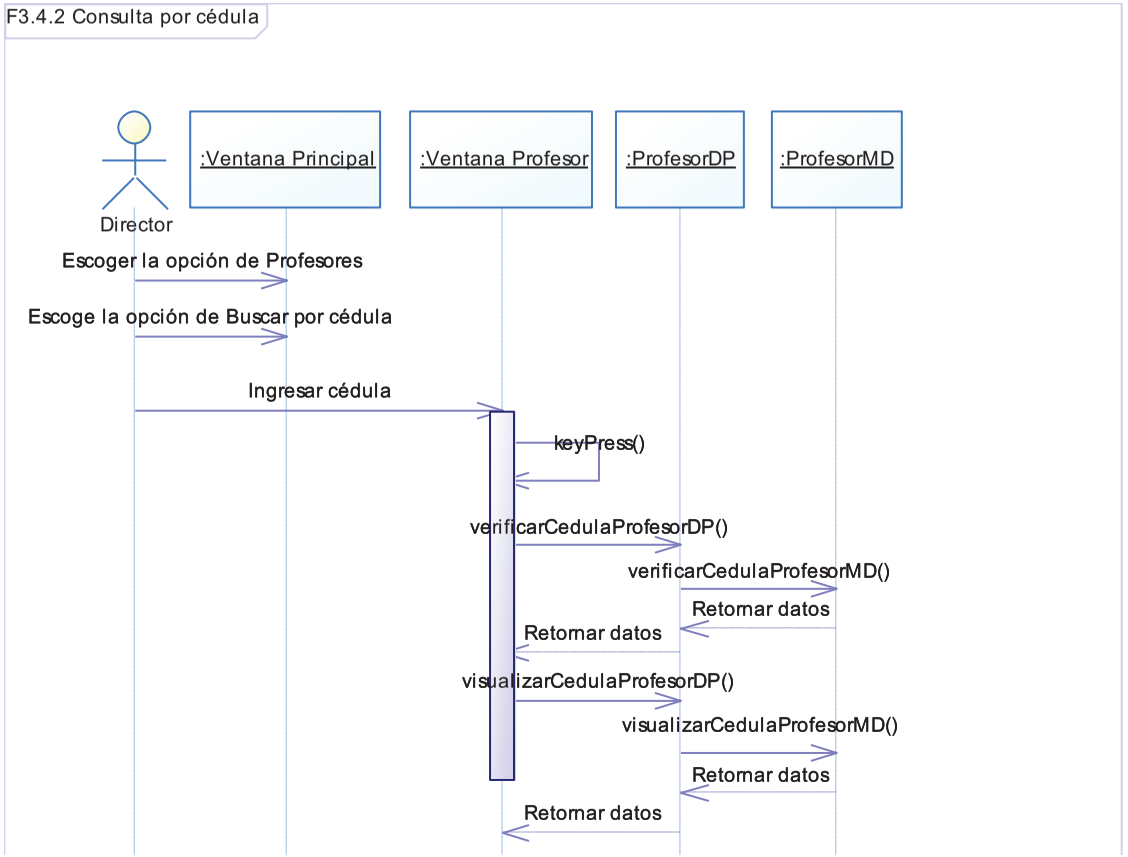

Escoger foto
Presiona botón Guardar
verificarDatos()
setNombre
setApellido
setSexo()
setLugarDeNaciomiento()
setFechaDeNacimiento()
setDireccion()
setTelefono()
setFechaInstitucion()
setFechaMagisterio()
setCodigoMagisterio()
setFoto()
agregarNuevoProfesorDP()
agregarNuevoProfesorMD()

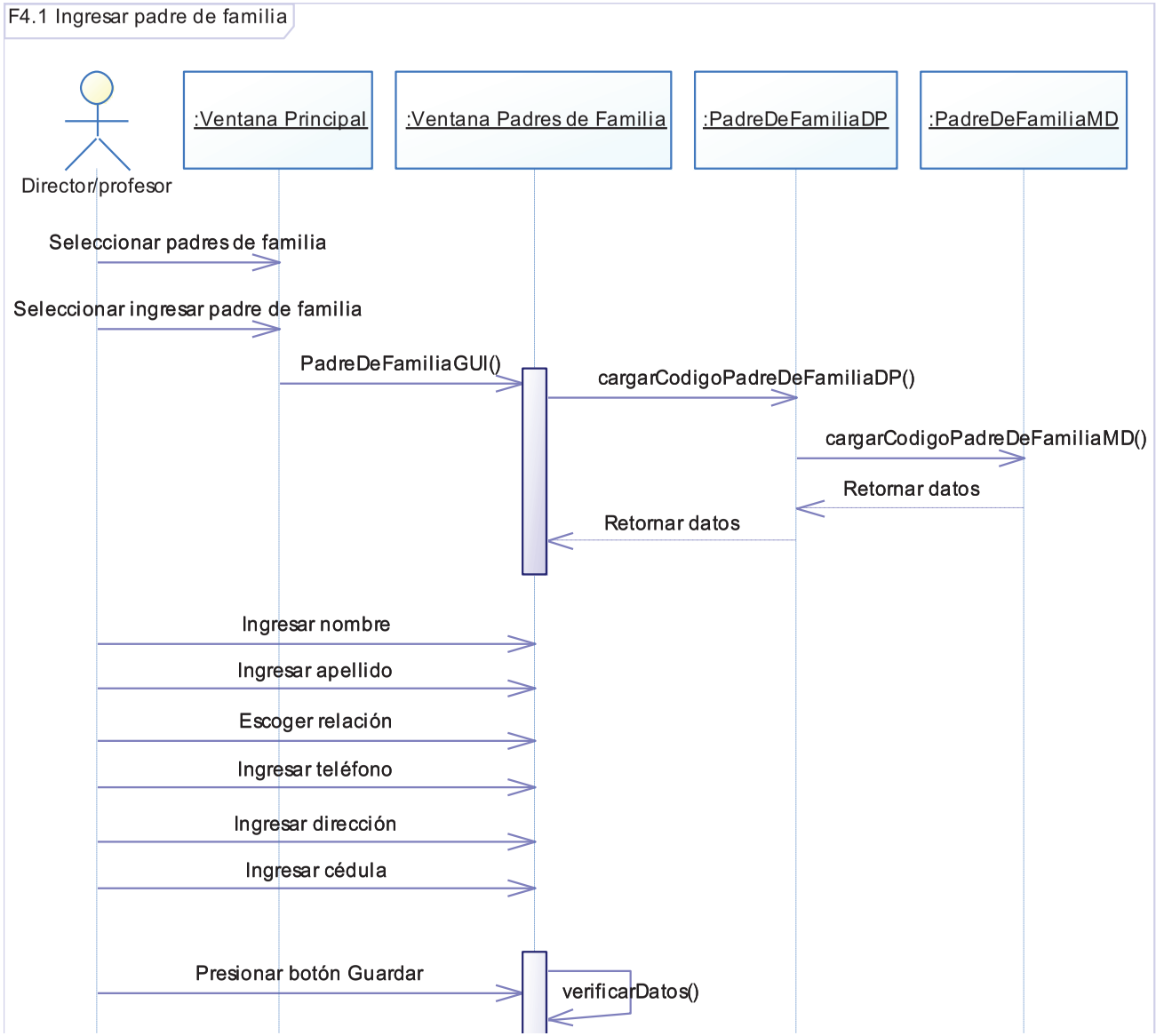

F3.2 Eliminar profesor
:Ventana Principal
:Ventana Profesor
:ProfesorDP
:ProfesorMD
Director
Escoger la opción de Profesores
Escoge la opción de Buscar por cédula
Ingresar cédula
keyPress()
verificarCedulaProfesorDP()
verificarCedulaProfesorMD()
Retornar datos
Retornar datos
visualizarCedulaProfesorDP()
visualizarCedulaProfesorMD()
Retornar datos
Retornar datos
Presiona botón Eliminar
msgConfirmarEliminacion()
Presinar botón Confirmar
eliminarProfesorDP()
elminarProfesorMD()

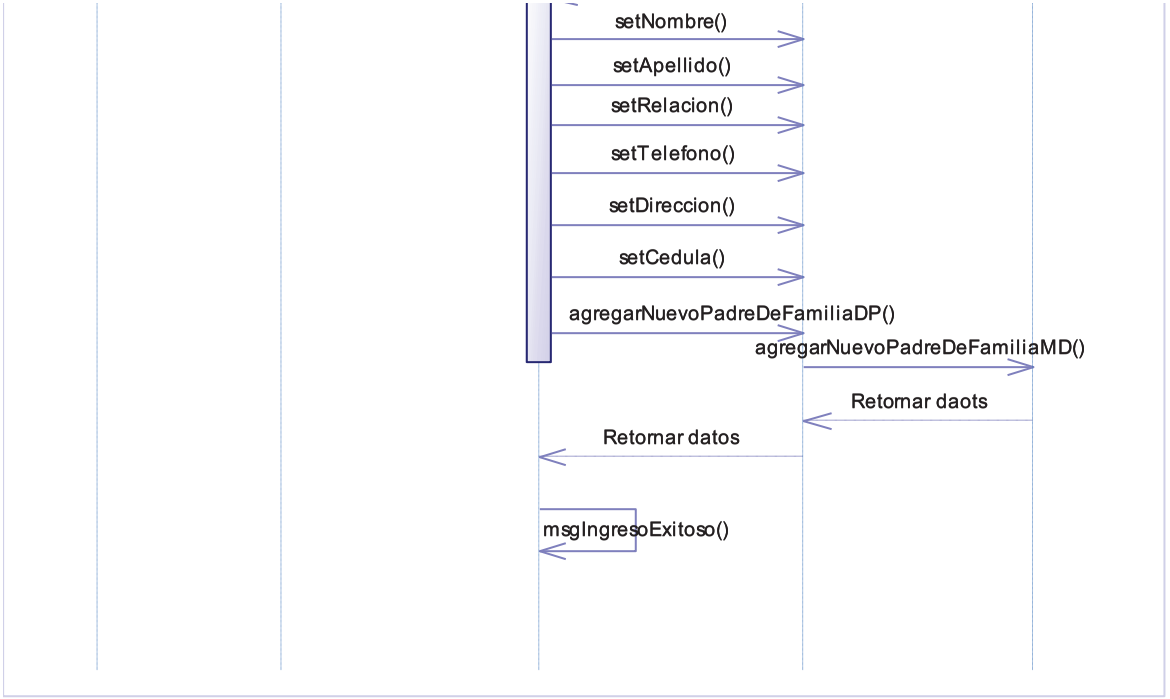

F3.3 Actualizar profesor
Director
:Ventana Principal
:Ventana Profesor
:ProfesorDP
:ProfesorMD
Escoger la opción de Profesores
Escoger busqueda por cédula
ProfesorGUI()
bloquearCampos()
Ingresar cédula
verificarCedulaProfesorDP()
verificarCedulaProfesorMD()
Retornar datos
Retornar datos
visualizarCedulaProfedorDP()
visualizarCedulaProfesorMD()
Retornar datos
Retornar datos
Presiona botón Modificar
desbloquearCampos()
Ingresar nombre
Ingresar apellido
Escoger sexo
Ingresar lugar de nacimiento
Escoger fecha de nacimiento
Ingresar dirección
Ingresar teléfono

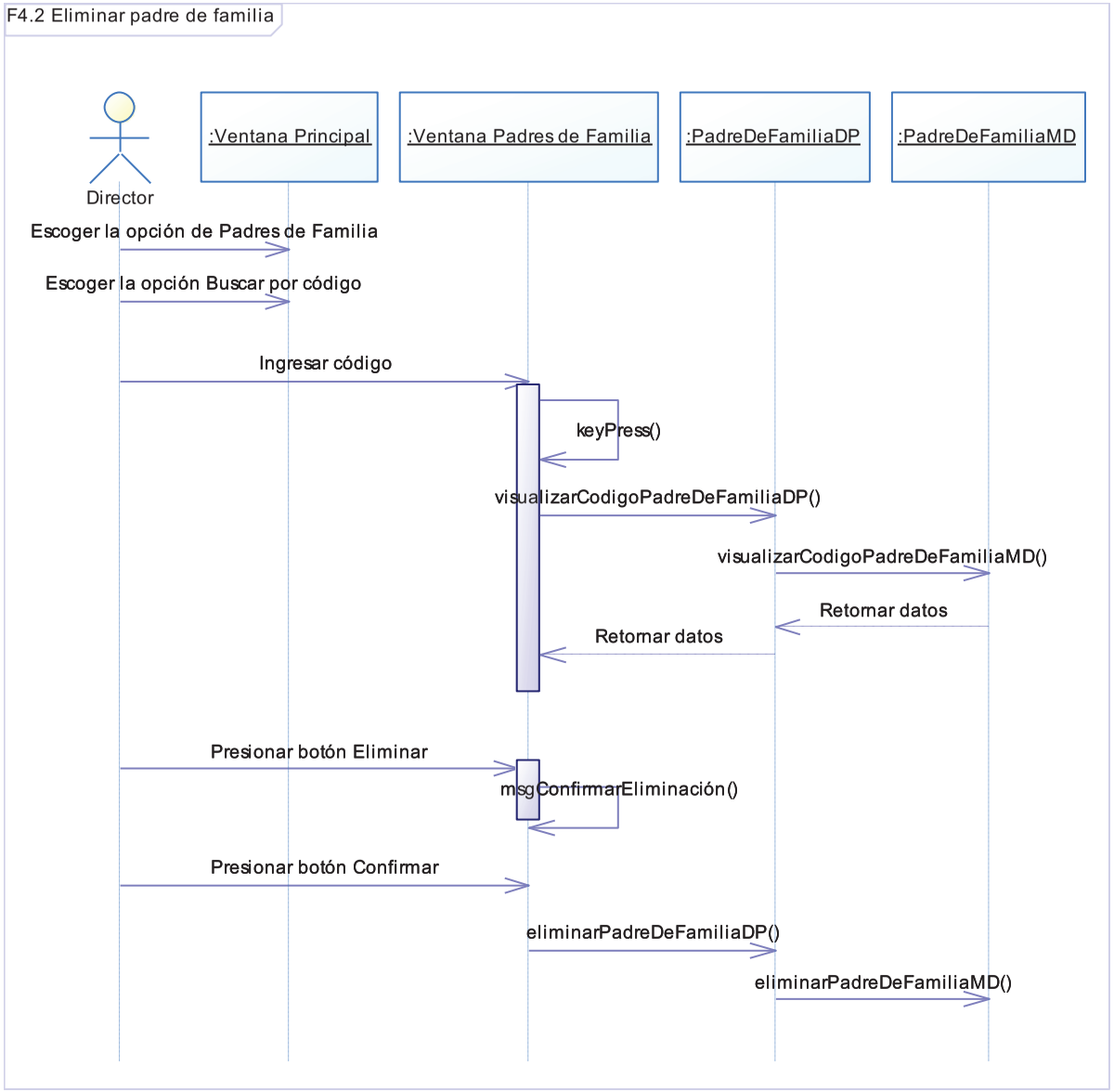

Escoger fecha de ingreso institución
Escoger fecha de ingreso magisterio
Ingresar código magisterio
Escoger foto
Presiona botón Guardar
verificarDatos()
setNombre
setApellido
setSexo()
setLugarDeNaciomiento()
setFechaDeNacimiento()
setDireccion()
setTelefono()
setFechaInstitucion()
setFechaMagisterio()
setCodigoMagisterio()
setFoto()
modificarProfesorDP()
modificarProfesorMD()

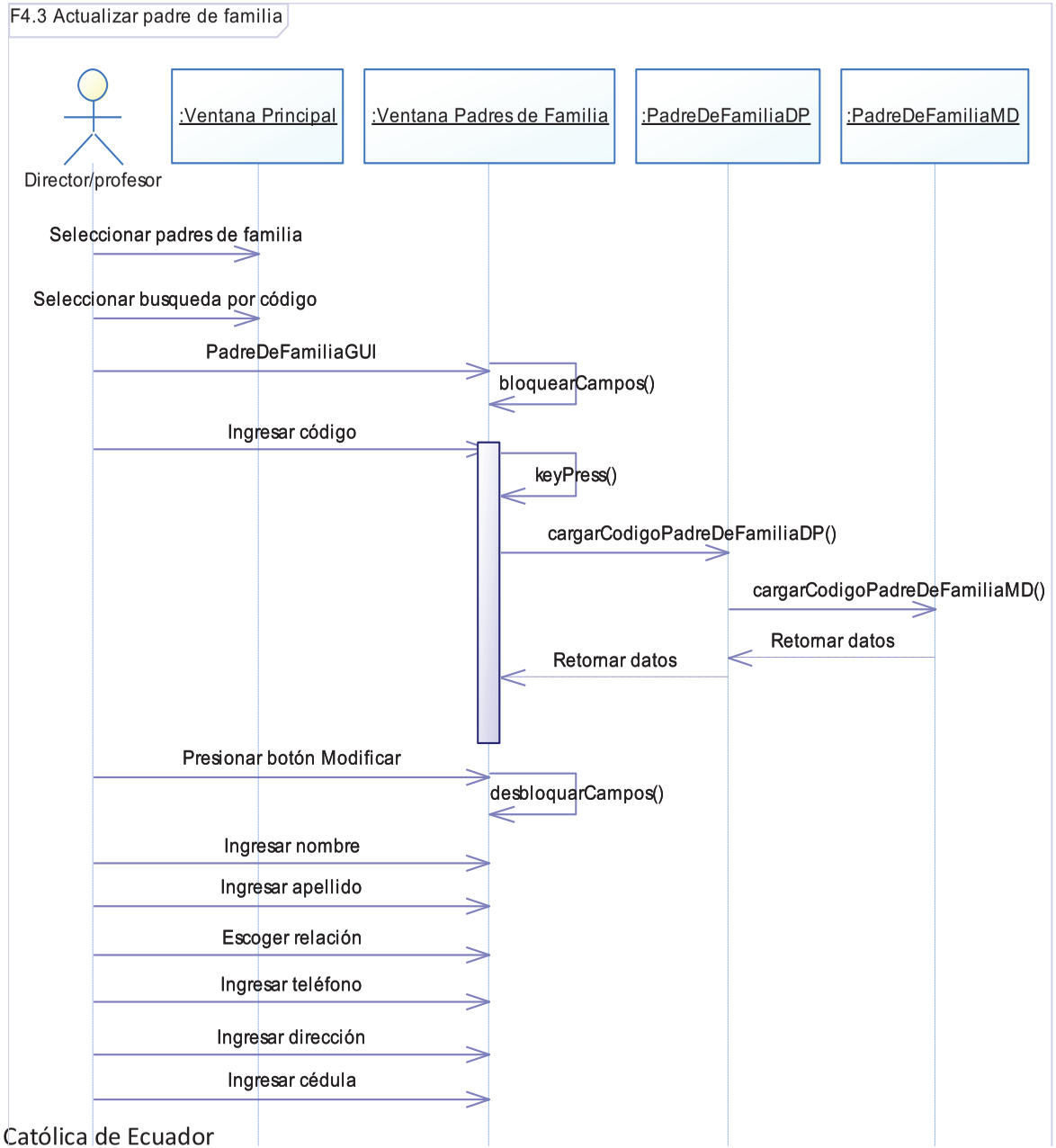

F3.4.1 Consulta general
:Ventana Principal
:Ventana Visualizar Profesor
:ProfesorDP
:ProfesorMD
Director/profesor
Seleccionar profesor
Seleccionar busqueda general
visualziarProfesor()
visualizarTodosProfesoresDP()
visualizarTodosProfesoresMD()
Retornar datos
Retornar datos

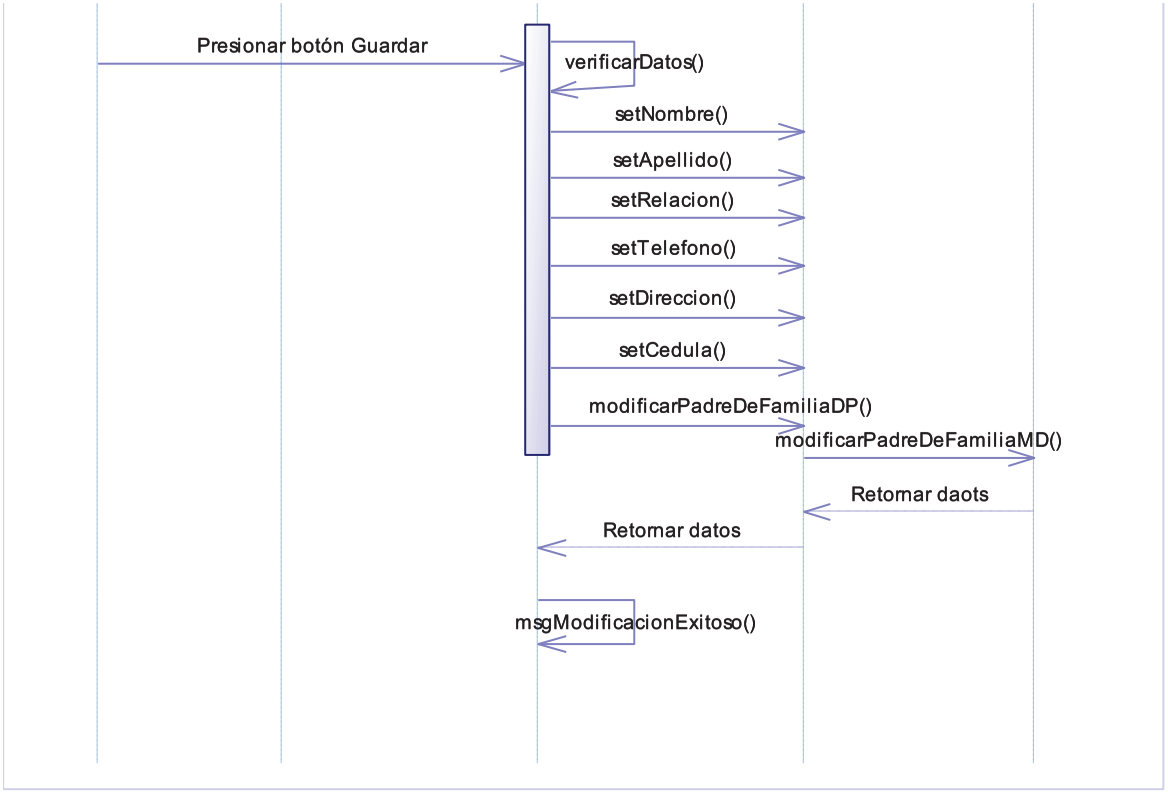

F3.4.2 Consulta por cédula
:Ventana Principal
:Ventana Profesor
:ProfesorDP
:ProfesorMD
Director
Escoger la opción de Profesores
Escoge la opción de Buscar por cédula
Ingresar cédula
keyPress()
verificarCedulaProfesorDP()
verificarCedulaProfesorMD()
Retornar datos
Retornar datos
visualizarCedulaProfesorDP()
visualizarCedulaProfesorMD()
Retornar datos
Retornar datos

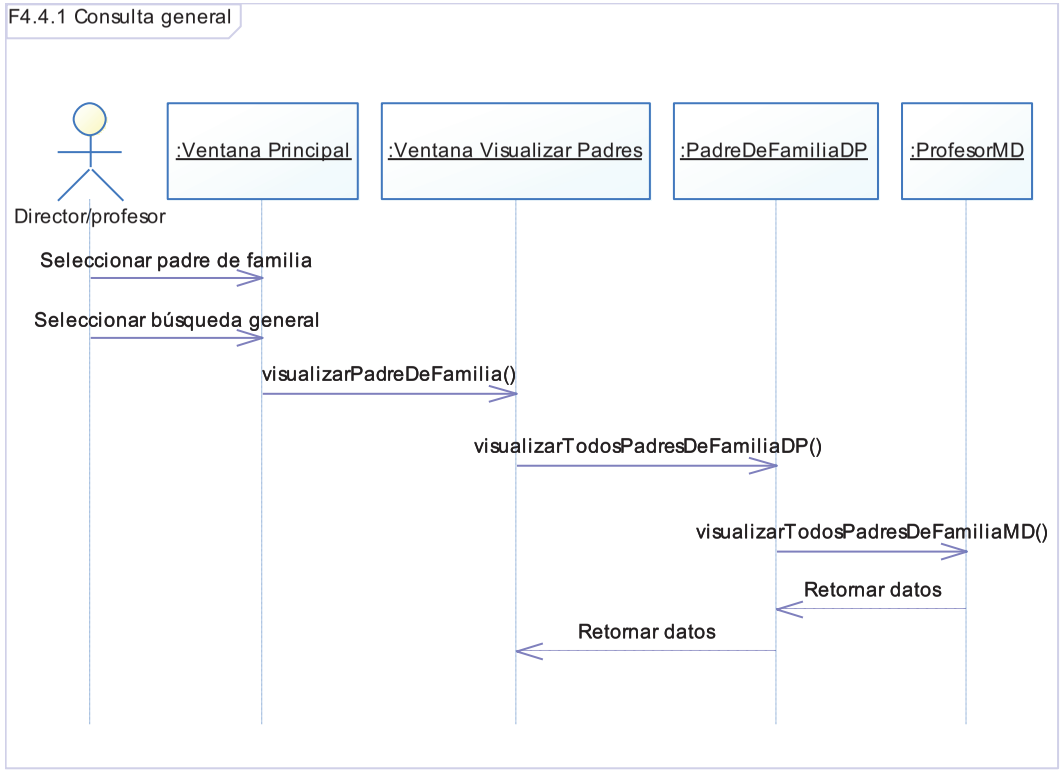

F4.1 Ingresar padre de familia
Director/profesor
:Ventana Principal
:Ventana Padres de Familia
:PadreDeFamiliaDP
:PadreDeFamiliaMD
Seleccionar padres de familia
Seleccionar ingresar padre de familia
PadreDeFamiliaGUI()
cargarCodigoPadreDeFamiliaDP()
cargarCodigoPadreDeFamiliaMD()
Retornar datos
Retornar datos
Ingresar nombre
Ingresar apellido
Escoger relación
Ingresar teléfono
Ingresar dirección
Ingresar cédula
Presionar botón Guardar
verificarDatos()

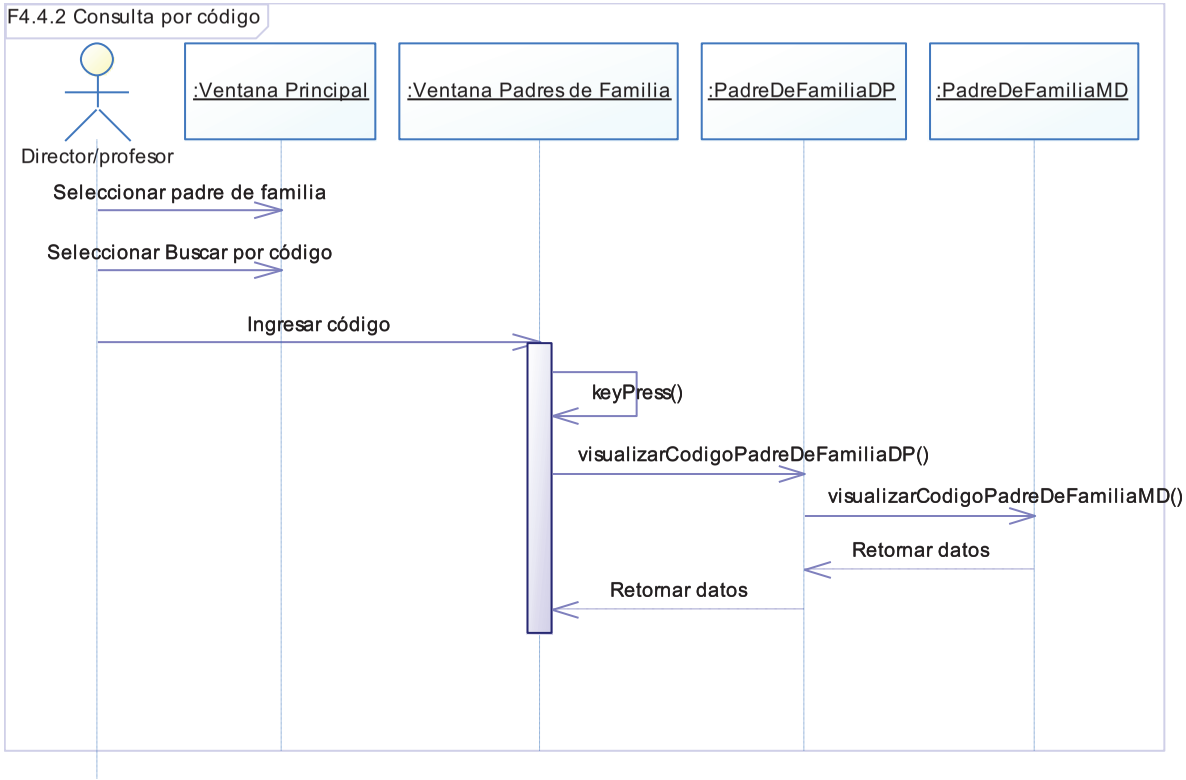

setNombre()
setApellido()
setRelacion()
setTelefono()
setDireccion()
setCedula()
agregarNuevoPadreDeFamiliaDP()
agregarNuevoPadreDeFamiliaMD()
Retornar daots
Retornar datos
msgIngresoExitoso()

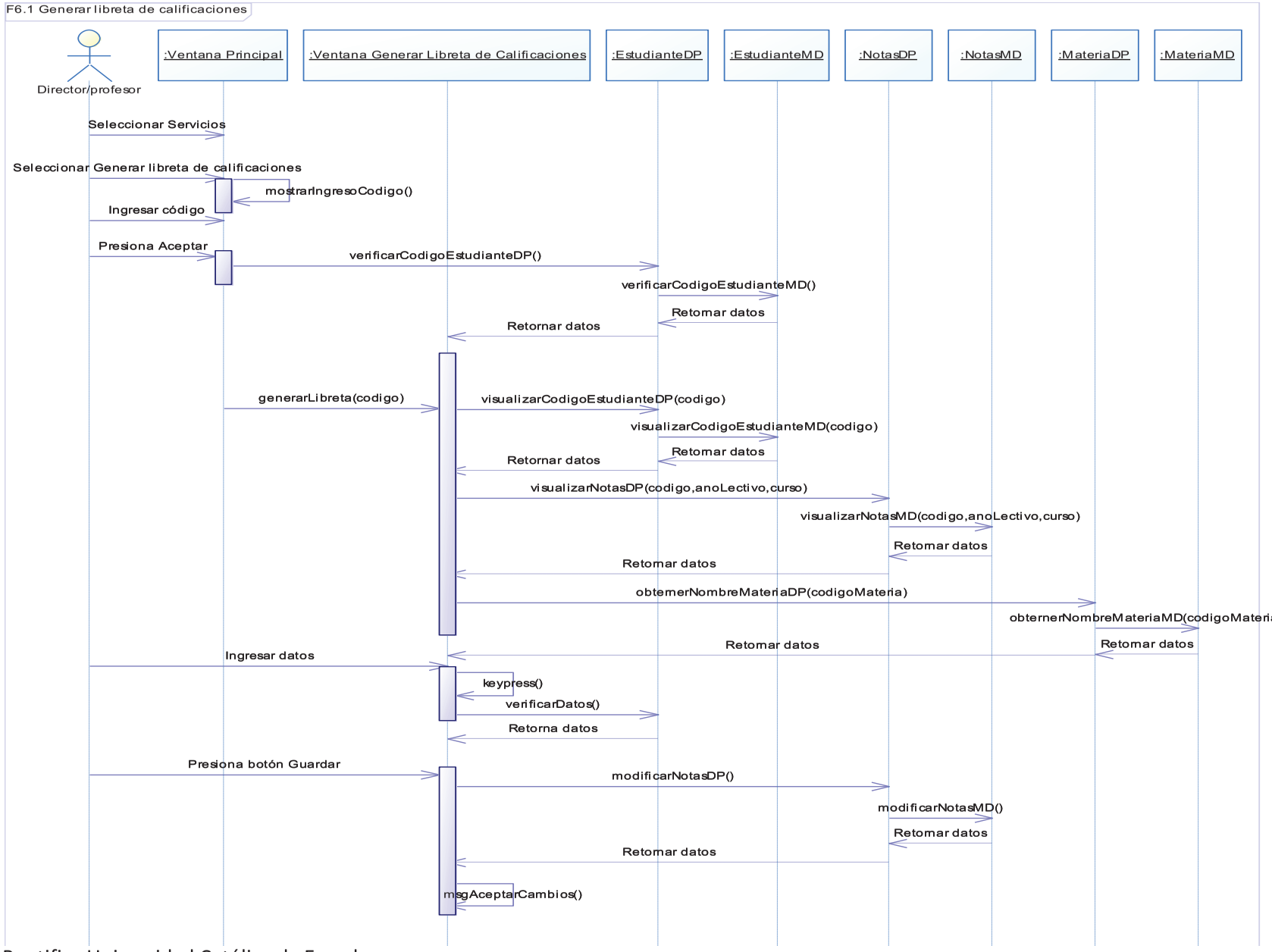

F4.2 Eliminar padre de familia
Director
:Ventana Principal
:Ventana Padres de Familia
:PadreDeFamiliaDP
:PadreDeFamiliaMD
Escoger la opción de Padres de Familia
Escoger la opción Buscar por código
Ingresar código
keyPress()
visualizarCodigoPadreDeFamiliaDP()
visualizarCodigoPadreDeFamiliaMD()
Retornar datos
Retornar datos
Presionar botón Eliminar
msgConfirmarEliminación()
Presionar botón Confirmar
eliminarPadreDeFamiliaDP()
eliminarPadreDeFamiliaMD()

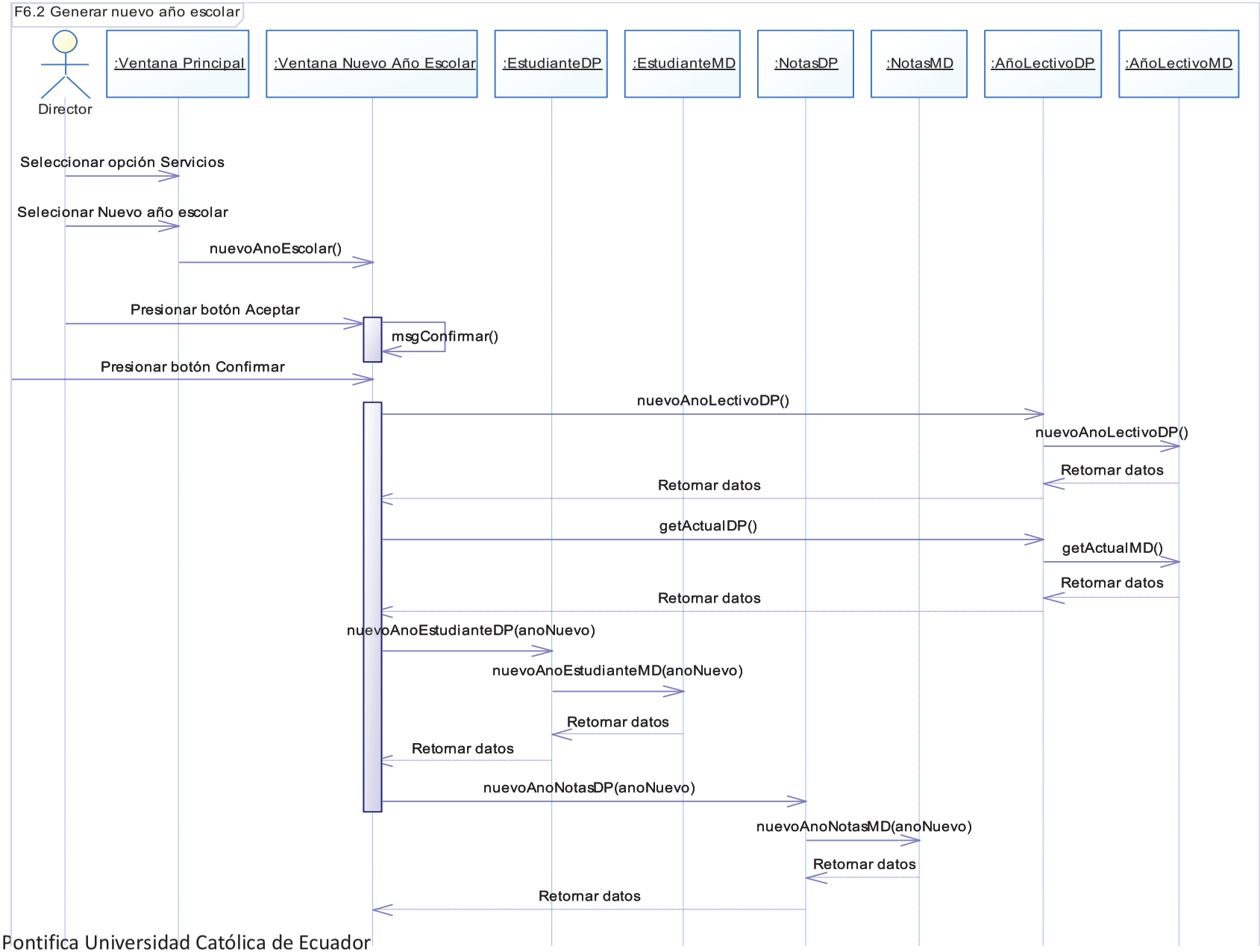

F4.3 Actualizar padre de familia
Director/profesor
:Ventana Principal
:Ventana Padres de Familia
:PadreDeFamiliaDP
:PadreDeFamiliaMD
Seleccionar padres de familia
Seleccionar busqueda por código
PadreDeFamiliaGUI
bloquearCampos()
Ingresar código
keyPress()
cargarCodigoPadreDeFamiliaDP()
cargarCodigoPadreDeFamiliaMD()
Retornar datos
Retornar datos
Presionar botón Modificar
desbloquarCampos()
Ingresar nombre
Ingresar apellido
Escoger relación
Ingresar teléfono
Ingresar dirección
Ingresar cédula

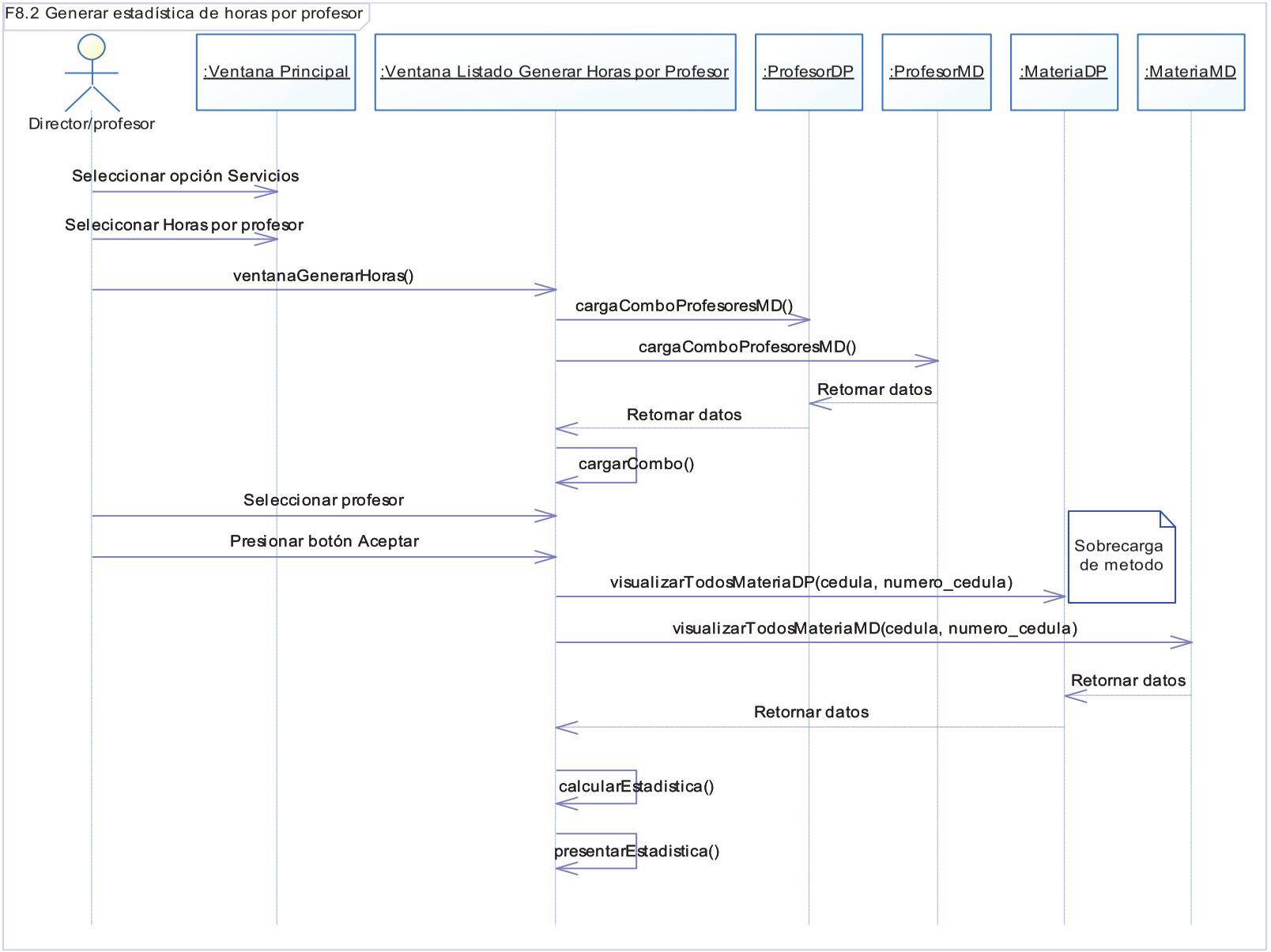

Presionar botón Guardar
verificarDatos()
setNombre()
setApellido()
setRelacion()
setTelefono()
setDireccion()
setCedula()
modificarPadreDeFamiliaDP()
modificarPadreDeFamiliaMD()
Retornar daots
Retornar datos
msgModificacionExitoso()


F4.4.1 Consulta general
:Ventana Principal
:Ventana Visualizar Padres
:PadreDeFamiliaDP
:ProfesorMD
Director/profesor
Seleccionar padre de familia
Seleccionar búsqueda general
visualizarPadreDeFamilia()
visualizarTodosPadresDeFamiliaDP()
visualizarTodosPadresDeFamiliaMD()
Retornar datos
Retornar datos


F4.4.2 Consulta por código
Director/profesor
:Ventana Principal
:Ventana Padres de Familia
:PadreDeFamiliaDP
:PadreDeFamiliaMD
Seleccionar padre de familia
Seleccionar Buscar por código
Ingresar código
keyPress()
visualizarCodigoPadreDeFamiliaDP()
visualizarCodigoPadreDeFamiliaMD()
Retornar datos
Retornar datos


F6.1 Generar libreta de calificaciones
Director/profesor
:Ventana Principal
:Ventana Generar Libreta de Calificaciones
:EstudianteDP
:EstudianteMD
:NotasDP
:NotasMD
:MateriaDP
:MateriaMD
Seleccionar Servicios
Seleccionar Generar libreta de calificaciones
mostrarIngresoCodigo()
Ingresar código
Presiona Aceptar
verificarCodigoEstudianteDP()
verificarCodigoEstudianteMD()
Retornar datos
Retornar datos
generarLibreta(codigo)
visualizarCodigoEstudianteDP(codigo)
visualizarCodigoEstudianteMD(codigo)
Retornar datos
Retornar datos
visualizarNotasDP(codigo,anoLectivo,curso)
visualizarNotasMD(codigo,anoLectivo,curso)
Retornar datos
Retornar datos
obtenerNombreMateriaDP(codigoMateria)
obtenerNombreMateriaMD(codigoMateria)
Retornar datos
Retornar datos
Ingresar datos
keypress()
verificarDatos()
Retorna datos
Presiona botón Guardar
modificarNotasDP()
modificarNotasMD()
Retornar datos
Retornar datos
msgAceptarCambios()


F6.2 Generar nuevo año escolar
Director
:Ventana Principal
:Ventana Nuevo Año Escolar
:EstudianteDP
:EstudianteMD
:NotasDP
:NotasMD
:AñoLectivoDP
:AñoLectivoMD
Seleccionar opción Servicios
Selecionar Nuevo año escolar
nuevoAnoEscolar()
Presionar botón Aceptar
msgConfirmar()
Presionar botón Confirmar
nuevoAnoLectivoDP()
nuevoAnoLectivoDP()
Retornar datos
Retornar datos
getActualDP()
getActualMD()
Retornar datos
Retornar datos
nuevoAnoEstudianteDP(anoNuevo)
nuevoAnoEstudianteMD(anoNuevo)
Retornar datos
Retornar datos
nuevoAnoNotasDP(anoNuevo)
nuevoAnoNotasMD(anoNuevo)
Retornar datos
Retornar datos


F8.2 Generar estadística de horas por profesor
Director/profesor
:Ventana Principal
:Ventana Listado Generar Horas por Profesor
:ProfesorDP
:ProfesorMD
:MateriaDP
:MateriaMD
Seleccionar opción Servicios
Seleciconar Horas por profesor
ventanaGenerarHoras()
cargaComboProfesoresMD()
cargaComboProfesoresMD()
Retornar datos
Retornar datos
cargarCombo()
Seleccionar profesor
Presionar botón Aceptar
Sobrecarga de metodo
visualizarTodosMateriaDP(cedula, numero_cedula)
visualizarTodosMateriaMD(cedula, numero_cedula)
Retornar datos
Retornar datos
calcularEstadistica()
presentarEstadistica()

3.3 Pruebas del sistema

Las pruebas del sistema son las que ayudan a definir a los desarrolladores como responde el programa a todas las situaciones que se le puedan presentar. Estas pruebas son creadas por los programadores para verificar que el sistema esté reaccionando de la manera que se espera y además probar su desenvolvimiento al mal uso del mismo, ya sea por fallas del usuario o del entorno (servicios, sistema operativo, etc.)

La ejecución de las pruebas fue avalada por el Lcdo. José Gabriel Guadamud Cantos, profesor de la Escuela Fiscal Mixta "Teodoro Wolf N°72" que rectifico la correcta funcionalidad del sistema, así como también acoto un conjunto de recomendaciones que permitirán mayor facilidad en el manejo de la herramienta propuesta con este trabajo.

Área de prueba: Validación de Usuario

Requerimientos

- Levantamiento de servicios

Actividad	Respuesta	Caso de Uso
Abrir el programa	Muestra ventana de ingreso	F1.
Validación de usuario Ingresar usuario Ingresar contraseña -Datos correctos ⟷ -Datos incorrectos ⟷	 Ingresa al sistema Mensaje de error	F1.1

Área de prueba: Administración de estudiantes

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Administración Estudiantes	Muestra opciones	F2.
Ingresar nuevo estudiante Ingresar los datos del estudiantes Comprobar los datos Presionar botón Guardar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de creación exitosa Mensaje de error	F2.1
Eliminación de estudiante Presionar Buscar código Ingresar código estudiante Presionar Eliminar Presionar botón Confirmar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de eliminación exitosa Mensaje de error	F2.2
Modificación de estudiante Presionar Buscar código Ingresar código estudiante Presionar Modificar Modificar datos Presionar botón Guardar -Datos correctos Datos incorrectos	Presenta resultados de operación. Mensaje de modificación exitosa Mensaje de error	F2.3
Consulta General Presionar Búsqueda General	Presenta el listado de estudiantes	F2.4

Área de prueba: Administración de profesores

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Profesores	Muestra opciones	F3.
<u>Ingresar nuevo profesor</u> Ingresar cédula Verificar Llenar formulario Presionar botón Guardar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de creación exitosa Mensaje de error	F3.1
<u>Eliminación de profesor</u> Presionar Buscar cédula Ingresar cédula Presionar Eliminar Presionar botón Confirmar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de eliminación exitosa Mensaje de error	F3.2
<u>Modificación de profesor</u> Presionar Buscar cédula Ingresar cédula Presionar Modificar Modificar campos Presionar botón Guardar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de modificación exitosa Mensaje de error	F3.3
<u>Consulta General</u> Presionar Búsqueda General	Presenta el listado de profesores	F3.4

Área de prueba: Administración de padres de familia

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Padres de familia	Muestra opciones	F4.
Ingresar nuevo Padres de familia Llenar formulario Presionar botón Guardar Verificar datos -Datos correctos -Datos incorrectos	 Presenta resultados de operación. Mensaje de creación exitosa Mensaje de error	F4.1
Eliminación de Padres de familia Presionar Buscar código Ingresar código Presionar Eliminar Presionar botón Confirmar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de eliminación exitosa Mensaje de error	F4.2
Modificación de Padres de familia Presionar Buscar código Ingresar código Presionar Modificar Modificar campos Presionar botón Guardar -Datos correctos -Datos incorrectos	 Presenta resultados de operación. Mensaje de modificación exitosa Mensaje de error	F4.3
Consulta General Presionar Búsqueda General	Presenta el listado de padres de familia	F4.4

Área de prueba: Administración de materias

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Materias	Muestra opciones	F5.
Ingresar nuevo Materias Llenar formulario Presionar botón Guardar Verificar datos -Datos correctos -Datos incorrectos	 Presenta resultados de operación. Mensaje de creación exitosa Mensaje de error	F5.1
Eliminación de Materias Presionar Buscar código Ingresar código Presionar Eliminar Presionar botón Confirmar -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de eliminación exitosa Mensaje de error	F5.2
Modificación de Materias Presionar Buscar código Ingresar código Presionar Modificar Modificar campos Presionar botón Guardar -Datos correctos -Datos incorrectos	 Presenta resultados de operación. Mensaje de modificación exitosa Mensaje de error	F5.3
Consulta General Presionar Búsqueda General	Presenta el listado de materias	F5.4

Área de prueba: Servicios

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Servicios	Muestra opciones	F6.
<u>Libreta de calificaciones</u> Ingresar código de estudiante Presionar botón Aceptar Llenar campos Verificar datos -Datos correctos -Datos incorrectos	Presenta resultados de operación. Mensaje de modificación exitosa Mensaje de error	F6.1
<u>Nuevo año escolar</u> Presiona el botón Nuevo Año Presiona botón confirmar cambio de año	Presenta resultados de operación. Mensaje de cambio exitosa	F6.2

Área de prueba: Impresiones

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Impresiones	Muestra opciones	F7.
Libreta de calificaciones: Curso Escoger el curso Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.1.1
Libreta de calificaciones: Código Ingresar el código del estudiante Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.1.2
Kardex: Curso Escoger el curso Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.2.1
Kardex: Código Ingresar el código del estudiante Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.2.2
Carnet: Curso Escoger el curso Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.3.1
Carnet: Código Ingresar el código del estudiante	Presenta resultados Mensaje de creación	F7.3.2

Presionar botón Aceptar	exitosa	
Horario Escoger el curso Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.4
Certificado anual Ingresar el código del estudiante Presionar botón Aceptar	Presenta resultados Mensaje de creación exitosa	F7.5

Área de prueba: Estadísticas

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Estadísticas	Muestra opciones	F8.
Horas por profesor Escoger el profesor de la lista	Presenta resultados de operación. Mensaje de creación exitosa	F8.1
Notas por curso Escoger el curso Presionar botón Aceptar	Presenta resultados de operación. Mensaje de creación exitosa	F8.2
Promedios generales Dar clic en Promedios generales	Presenta resultados de operación. Mensaje de creación exitosa	F8.3

Área de prueba: Respaldo de BDD

Requerimientos

- Levantamiento de servicios
- Ingreso al sistema

Actividad	Respuesta	Caso de Uso
Seleccionar Respaldo de BDD	Muestra opciones	F9.
Respaldo Actual Dar clic en la opción Respaldo Actual Escoger la dirección donde se va a guardar el respaldo Ingresar el nombre del respaldo	Presenta resultados de operación. Mensaje de creación exitosa	F9.1

CAPITULO IV: PRUEBAS E IMPLEMENTACIÓN

En este capítulo se explicará los estándares que se utilizaron para programar en el desarrollo del software y además las pruebas que se le realizaron a los mismos, con el fin de obtener un producto de calidad.

4.1 Pruebas de usuarios

Estas pruebas se las realizan para obtener el punto de vista del lado del usuario, con lo que obtenemos una retroalimentación rápida o concisa de cómo se está desarrollando el producto en cada iteración del mismo.

Cuando se consigue resultados positivos en un iteración, se logra la base para repetirlas en las demás El objetivo es lograr un 100% de efectividad en cada iteración.

PRUEBA DE USUARIO	
Nombre de prueba	Ingreso al sistema
Referencia #	1
Iteración	1
Funcionamiento critico	Si
¿Qué hace esta prueba?	Prueba la ventana de ingreso, validación del usuario y de la contraseña.
Categoría	Manejo de usuarios (F1)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Ingreso al sistema	Ingresar usuario y contraseña, Aceptar	U:Admin C:12345	Ingreso fallido
2	Ingreso al sistema	Ingresar usuario y contraseña, Aceptar	U:admin C:12345	Ingreso exitoso
3	Ingreso al sistema	Ingresar usuario y contraseña, Aceptar	U: (vacío) C:12345	Ingreso fallido
4	Ingreso al sistema	Ingresar usuario y contraseña, Aceptar	U:!@#$%^&* C:21398	Ingreso fallido
5	Ingreso al sistema	Ingresar usuario y contraseña, Aceptar	U:admin C: (vacío)	Ingreso fallido

PRUEBA DE USUARIO	
Nombre de prueba	Matriculación
Referencia #	2
Iteración	1
Funcionamiento critico	Si
¿Qué hace esta prueba?	Prueba de ingreso de estudiantes, carga de foto de estudiante, datos requeridos completos, modificación/eliminación de campos
Categoría	Administración de Estudiantes (F2)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Ingreso de nuevo estudiante	Ingresar datos requeridos incompletos	Estudiante existente en la institución	Faltan datos por ingresar
2	Ingreso de nuevo estudiante	Ingresar datos requeridos completos sin foto	Estudiante existente en la institución	Estudiante creado correctamente
3	Ingreso de nuevo estudiante	Ingresar datos requeridos completos con foto	Estudiante existente en la institución	Estudiante creado correctamente
4	Ingreso de nuevo estudiante	Cargar foto excediendo límite de tamaño.	Foto de prueba de 1MB(limite 0.3MB)	Foto demasiado grande. No cargada
5	Buscar código	Modificar estudiante con datos incompletos	Estudiante existente en la institución	Faltan datos por ingresar

6	Buscar código	Modificar estudiante con datos completos	Estudiante existente en la institución	Estudiantes modificado correctamente
7	Buscar código	Eliminar estudiante	Código de estudiante de prueba	Estudiantes eliminado correctamente
8	Búsqueda general	Buscar estudiantes por filtro de nombre	Estudiante: "Car"	Estudiante encontrado
9	Búsqueda general	Buscar estudiantes por filtro de curso	Curso: 1	Estudiantes encontrados

PRUEBA DE USUARIO	
Nombre de prueba	Ingreso profesores
Referencia #	3
Iteración	1
Funcionamiento critico	Si
¿Qué hace esta prueba?	Prueba de ingreso de profesores, carga de foto de profesores, datos requeridos completos, modificación/eliminación de campos
Categoría	Administración de Profesores (F3)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Ingreso de nuevo profesor	Ingresar datos requeridos incompletos	Profesor existente en la institución	Faltan datos por ingresar
2	Ingreso de nuevo profesor	Ingresar datos requeridos completos sin foto	Profesor existente en la institución	Profesor creado correctamente
3	Ingreso de nuevo profesor	Ingresar datos requeridos completos con foto	Profesor existente en la institución	Profesor creado correctamente
4	Ingreso de nuevo profesor	Cargar foto excediendo límite de tamaño.	Foto de prueba de 1MB(limite 0.3MB)	Foto demasiado grande. No cargada
5	Ingreso de nuevo profesor	Ingresar datos requeridos con cédula ya utilizada.	Profesor existente en la institución	Cédula ya asignada a un profesor.

6	Buscar código	Modificar profesor con datos incompletos	Profesor existente en la institución	Faltan datos por ingresar
7	Buscar código	Modificar profesor con datos completos	Profesor existente en la institución	Profesor modificado correctamente
8	Buscar código	Eliminar profesor	Cédula de profesor de prueba	Profesor eliminado correctamente
9	Búsqueda general	Buscar profesor por filtro de nombre	Profesor: "Ped"	Profesor encontrado

PRUEBA DE USUARIO	
Nombre de prueba	Ingreso padres de familia
Referencia #	4
Iteración	1
Funcionamiento critico	No
¿Qué hace esta prueba?	Prueba de ingreso de padres de familia, datos requeridos completos, modificación/eliminación de campos
Categoría	Administración de Padres de familia (F4)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Ingreso de nuevo padres de familia	Ingresar datos requeridos incompletos	Padre de familia existente en la institución	Faltan datos por ingresar
2	Ingreso de nuevo padres de familia	Ingresar datos requeridos completos	Padres de familia existente en la institución	Padres de familia creado correctamente
3	Buscar código	Modificar padres de familia con datos incompletos	Padres de familia existente en la institución	Faltan datos por ingresar
4	Buscar código	Modificar padres de familia con datos completos	Padres de familia existente en la institución	Padres de familia modificado correctamente
5	Buscar código	Eliminar padres de familia	Código de padres de	Padres de familia

			familia de prueba	eliminado correctamente
6	Búsqueda general	Buscar padres de familia por filtro de nombre	Padres de familia: "Mari"	Padres de familia encontrado

PRUEBA DE USUARIO	
Nombre de prueba	Ingreso materias
Referencia #	5
Iteración	1
Funcionamiento critico	Si
¿Qué hace esta prueba?	Prueba de ingreso de materias, datos requeridos completos, modificación/eliminación de campos
Categoría	Administración de Materias (F5)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Ingreso de nueva materia	Ingresar datos requeridos incompletos	Materia existente en la institución	Faltan datos por ingresar
2	Ingreso de nueva materia	Ingresar datos requeridos completos	Materia existente en la institución	Materia creada correctamente
3	Ingreso de nueva materia	Ingresar combinación de nombre de materia y curso asignado, ya existente.	Materia existente en la institución	Materia ya existe. No se ha podido guardar
4	Buscar código	Modificar materia con datos incompletos	Materia existente en la institución	Faltan datos por ingresar
5	Buscar código	Modificar materia con datos	Materia existente en la	Materia modificada

		completos	institución	correctamente
6	Buscar código	Eliminar materia	Código de materia de prueba	Materia eliminada correctamente
7	Búsqueda general	Buscar materia por filtro de nombre	Materia: "Mate"	Materia encontrada
8	Búsqueda general	Buscar materia por filtro de curso	Curso: 4	Materia encontrada

PRUEBA DE USUARIO	
Nombre de prueba	Servicios
Referencia #	6
Iteración	2
Funcionamiento critico	Si
¿Qué hace esta prueba?	Modificación de libreta, eliminación de notas, cambio de año escolar
Categoría	Servicios (F6)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Creación de nueva libreta	Ingresar notas de prueba	Libreta de calificaciones de prueba	Libreta modificada correctamente
2	Creación de nueva libreta	Ingresar valores superiores a máxima nota (20)	21,20.00009,57,20.01	El valor debe ser igual o inferior a 20
3	Creación de nueva libreta	Ingresar letras en campo de notas	Ea,5t,4s	Valor no valido.
4	Creación de nueva libreta	Eliminar notas antiguas	Libreta de calificaciones de prueba	Libreta modificada correctamente
5	Cambio de año escolar	Cambiar del año escolar presente a uno nuevo.	Cambio del 2012-2013 al 2013-2014	Cambio exitoso.

PRUEBA DE USUARIO	
Nombre de prueba	Impresiones
Referencia #	7
Iteración	3
Funcionamiento critico	No
¿Qué hace esta prueba?	Impresiones de todos los documentos que emite la institución.
Categoría	Impresiones (F7)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Impresión de libreta de calificaciones: Curso	Escoger un curso	Curso: 1 Curso: 2	Libretas de calificaciones creada
2	Impresión de libreta de calificaciones: Código	Ingresar código	Código: 0001 Código: 0002 Código: 0003	Libreta de calificaciones creada
3	Impresión de libreta de calificaciones: Código	Ingresar código	Código: 1414	Estudiante no existe
4	Impresión de Kardex: Curso	Escoger un curso	Curso: 2 Curso: 3	Kardex creado
5	Impresión de Kardex:	Ingresar código	Código: 0004	Kardex creado

	Código		Código: 0005 Código: 0006	
6	Impresión de Kardex: Código	Ingresar código	Código: 1415	Estudiante no existe
7	Impresión de carnet: Curso	Escoger un curso	Curso: 3 Curso: 4	Carnet creado
8	Impresión de carnet: Código	Ingresar código	Código: 0011 Código: 0012 Código: 0013	Carnet creado
9	Impresión de carnet: Código	Ingresar código	Código: 5484	Estudiante no existe
10	Impresión de horario	Escoger un curso	Curso: 6 Curso: 7	Horario creado
11	Impresión de certificado anual	Ingresar código	Código: 0010 Código: 0008 Código: 0009	Certificado anual creado
12	Impresión de certificado anual	Ingresar código	Código: 9999	Estudiante no existe

PRUEBA DE USUARIO	
Nombre de prueba	Estadísticas
Referencia #	8
Iteración	2
Funcionamiento critico	No
¿Qué hace esta prueba?	Creación de estadísticas.
Categoría	Estadísticas (F8)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Creación de estadística: Horas por profesor	Escoger profesor de la lista	Profesor de prueba	Estadística creada
2	Creación de estadística: Horas por profesor	Buscar materia por filtro de nombre	Materia: "Mate"	Estadística creada
3	Creación de estadística: Notas por curso	Escoger curso	Curso: 1	Estadística creada
4	Creación de estadística: Promedios generales	Escoger la opción Promedios generales	Libretas creadas en pasos anteriores.	Estadística creada

PRUEBA DE USUARIO	
Nombre de prueba	Respaldo de base de datos
Referencia #	9
Iteración	2
Funcionamiento critico	No
¿Qué hace esta prueba?	Respaldar la base de datos actual en un archivo transportable.
Categoría	Respaldo BDD (F9)
Prerrequisito	Usuario creado con permisos adecuados, conexión a BDD

Avalado por Lcdo. José Gabriel Guadamud Cantos, 12/03/2012

Prueba	Pantalla	Acción	Datos de entrada	Resultado obtenido
1	Respaldo actual de BDD	Ingresar la ubicación y el nombre de la base de respaldo.	Nombre de Archivo: "Respaldo" Ubicación: Escritorio	Respaldo creado con nombre asignado y ubicación correcta
2	Respaldo actual de BDD	Escoger la opción y cancelar el respaldo	Nombre de Archivo: "Respaldo2" Ubicación: Escritorio	Respaldo no creado con mensaje de notificación.

4.2 Modificaciones finales

A pesar de presentar continuos lanzamientos del software, a través de iteraciones; existieron modificaciones que se presentaron con la inclusión de todos los usuarios (profesores) que iban a utilizar el producto y por entes externos a los cuales se debe regir la institución. A continuación se detallan los cambios presentados:

#	Modificación	Razón	Relevancia	Duración
1	Aumento de campo "Ingreso al magisterio" en datos de Profesor	Información necesaria	Mínima	3 horas
2	Cambio de etiqueta de "Curso" a "Año Lectivo" en todas las pantallas.	Estandarización de documentos	Mínima	2 horas
3	Cambio de etiqueta de Notas 1,2,3 en Trimestre 1,2,3	Estandarización de documentos	Mínima	2 horas
4	Cambio de etiqueta de "Aprovechamiento" a "Desempeño Académico" en pantallas y documentos.	Requisito del Ministerio de Educación	Mínima	3 horas
5	Cambio de 2 decimales a 3 decimales en cálculos de promedios.	Requisito del Ministerio de Educación	Media	2 horas
6	Nuevo documento para impresión "Certificado Anual"	Requisito del Ministerio de Educación	Media	4 horas
7	Corrección de consistencia, no permitir materias repetidas para un mismo curso	Seguridad	Media	3 horas
8	Permitir ser representante a los 3 tipos de relaciones de padre de familia	Estandarización de documentos	Mínima	1 hora

4.3 Implementación del sistema

4.3.1 Estándares de codificación

Para que el código sea entendible para los colaboradores o para una futura versión es necesario que se maneje un estándar de programación, lo que ayuda a realizar cambios, optimizaciones, reutilizaciones y mantenimiento de una manera mucho más rápida.

Los siguientes son los estándares utilizados a través del desarrollo del programa:

Estructura de un archivo	
Comentario inicial Se debe comenzar con un comentario que tenga el nombre de la clases, información sobre la versión del código, el autor , la fecha de modificación, y proyecto al cual pertenece	* * generarLibreta.java * * Ultima Modificación * Autor: Vicente Mendoza * Fecha: 05-enero-2012 * * Proyecto: Tesis Teodoro Wolf */
Declaración de paquetes La primera línea no comentada de un archivo debe ser la declaración a que paquete pertenece	package servicios.procesos;

Sentencias de importación
Después de la declaración de paquetes se deben realizar las importaciones. Las cuales deben ser explicitas. No se deben utilizar las importaciones de tipo "package" ya que dificultan la asociación y entendimiento del código, a menos que se especifique su uso.
	import java.io.*; // BufferedReader, PrintWriter, FileInputStream, File
	import java.util.ArrayList;
	import java.util.Map;
	import java.util.HashMap

Largo de líneas Evitar tener líneas de código con más de 80	

caracteres, simplemente si se da el caso dividirla en dos o más líneas.	

Nomenclatura	
Nombre de paquetes Todo el nombre del paquete debe ser en minúsculas.	com.janox.customers, com.janox.customers.data
Nombre de clases e interfaces La primera letra de cada palabra debe ser en mayúsculas. Deben ser palabras completas o abreviaturas para entender de que se tratan si abrir su contenido.	Item ManejoDatos
Métodos y variables La primera letra debe ser en minúsculas, para las siguientes palabras se comienza con mayúsculas.	agregarItem() setNombre() getApellido()
Constantes Aunque no se recomienda su utilización, se maneja el estándar de todas las letras en mayúsculas.	MAX_ITEM_NUMBER DEFAULT_PREFIX

Declaración de variables	
Se debe realizar una declaración en cada línea, lo que permite poner un comentario explicativo.	String hora; //hora de entrada
Dar un valor inicial a la variable. Siempre hacer las declaraciones en el inicio de un bloque.	String hora=0;
Darle a las variables un nombre significativo.	
Utilizar siempre corchetes ({}) aunque el código solo sea de una línea y se pueda obviar.	if(hora>5){ hora=hora-1;}
Si se quiere ignorar una excepción indicarlo explícitamente.	try { } catch (Exception ignorada) {}

4.3.2 Código fuente

El código fuente junto con los demás componentes de este trabajo, debido a su tamaño, se encuentra almacenada en el CD que se anexa.

4.4 Mantenimiento del sistema

El producto siempre se mantuvo funcionando, mientras se iban realizando las nuevas iteraciones, se las integraba al software para mantenerlo en producción. Para realizar el mantenimiento post producción del software se acordaron 3 reuniones en puntos críticos de uso del producto, de acuerdo a si existía o no alguna incidencia.

#	Fecha	Motivo
1	02-06 de abril 2012	Terminación de periodo de matriculación.- Verificación de ingreso normal de toda la nómina de estudiantes y de profesores.
2	18-25 de junio 2012	Fin de primer trimestre estudiantil.- Verificación de impresión de documentos de la institución.
3	02-11 de enero 2013	Terminación del año escolar.- Cambio de año escolar, generación de nuevas libretas escolares.

CAPITULO V: CONCLUSIONES Y RECOMENDACIONES

En este capítulo se presentan los resultados que se obtuvieron, durante todo el desarrollo del proyecto y luego de la entrega del mismo a través de las conclusiones y recomendaciones.

5.1 Conclusiones

Una vez que se ha terminado el proyecto pasando por el proceso de levantamiento de requerimientos, análisis de los mismos, planificación, desarrollo del proyecto, instalación del producto entre otros pasos, se dio por terminado todos los objetivos que se establecieron en un principio. Es así que se llega a las siguientes conclusiones:

- La metodología ágil Xtreme Programming, fue una elección acertada ya que contribuyó al progreso del software en todos sus aspectos. Si bien tuvo que adaptarse a este proyecto específico, se pudo realizar cambios de gran importancia en poco tiempo, además de contar con la documentación necesaria para evaluar, planificar y desarrollar el producto.
- La comunicación y contacto con el cliente fue pieza fundamental ya que contribuyó con todos los documentos necesarios que se automatice su creación, su tiempo y decisiones fueron vitales en el desarrollo, ya que aparte de definir lo que requería, informaba de las reglas a las que debían atenerse de parte de organismo externos (Ministerio de Educación).
- El desarrollador de software debe tener un conocimiento avanzado de las herramientas a utilizar durante la realización del producto, ya que es fundamental para la resolución de problemas, toma de decisiones, corrección de errores, etc. Las herramientas además deben ser compatibles entre ellas para no ocasionar ningún inconveniente en la puesta en producción.
- El producto creado, aunque se cataloga como software a medida y fue diseñado para la Escuela Teodoro Wolf N°72, puede ser reutilizado en otras instituciones con las variantes respectivas de dichas instituciones.
- Los estándares de codificación ayudan significativamente a la comprensión del mismo, lo que facilita la adaptación de nuevos módulos, mantenimiento u

optimización; ya que no siempre el desarrollador que comienza el proyecto es el que continúa con él.

- La elección de las herramientas utilizadas, tanto para el desarrollo (Netbeans) y la de base de datos (MySQL), fueron las correctas ya que el entorno de trabajo es amigable y de fácil entendimiento. Además se tenía un conocimiento previo de dichas herramientas lo que facilito su uso. Por último, ambos manejaban una documentación adecuada y bastantes foros de ayuda para resolver problemas.
- El proyecto se concluyó con éxito, instalando el software a la Escuela Teodoro Wolf N°72, capacitando al personal de la institución y entregando los manuales respectivos. Con el aplicativo la institución va a automatizar el registro de matrículas, las cálculos de notas, la impresión de documentos. Además de tener una base de datos consistente de estudiantes, profesores y padres de familia.

5.2 Recomendaciones

Se proponen algunas medidas sobre la utilización y desarrollo del software. Así como también sobre las herramientas y las metodologías usadas con el fin de mejorar lo anteriormente realizado.

- Se dejó propuesto la adecuación de un nuevo módulo, que se basa en la visualización de las notas obtenidas hacia los estudiantes, mediante un código de barras colocado en los carnets. Se espera su implementación en una segunda versión del programa.
- Utilizar en la medida que se posible, herramientas de software libre, ya que se tiene por sentado, que se va a obtener todas las ventajas de esté, como lo son distribución, mejoramiento, compatibilidad, uso gratuito, soporte, etc.
- Para proyectos no tan grandes se recomienda utilizar metodologías agiles ya que se concentran en lograr los objetivos, si tener tanta burocracia. Lo que las hace ideales en proyectos pequeños y medianos con poco tiempo, dinero y personal.
- Se debe planificar tiempos para los imprevistos, ya que siempre se presentan en cualquier proyecto, tratando de no dañar cronogramas establecidos.

- El usuario final debe leer al pie de la letra el manual de usuario, para poder utilizar el aplicativo, ya que su mala utilización puede conllevar a su mal funcionamiento. Para esto el manual se ha realizado de la forma más clara posible, de la misma forma el Sistema Académico Natlog fue pensado de la manera más intuitiva posible.
- Planificar reuniones con el cliente, ya que este es el que decide si lo se está haciendo está acorde a sus necesidades.
- Realizar la automatización de procesos de manera escalada ya que es difícil integrar todas las actividades que realiza la institución en un solo paso.
- Tomar en cuenta aspectos externos al proyecto que no dependen ni del cliente ni de los desarrolladores, como lo son medidas gubernamentales, leyes, etc. Estas influyeron en el proyecto ya que el Ministerio de Educación les pedía cambios en la institución que representaron contratiempos que debieron ser cubiertos de una u otra manera, como por ejemplo se hicieron 3 modelos de libretas para que puedan ser aprobados.
- Reutilizar el código, es una manera muy eficiente de sacar provecho a lo que ya se ha realizado, en el aplicativo se reutilizan muchas ventanas lo que ahorra recursos.
- En nuevas versiones se plantean además de la información de padres de familia, también la actividades que realizan en la escuela; un módulo de recursos económicos con las donaciones , equipamiento y capital económico que se recibe de diferentes entidades, registros sobre la salud de los estudiantes en la institución se vacuna y se les brinda pastillas a los estudiantes.

BIBLIOGRAFIA

- Desarrollo de software. Programación extrema (eXtreme Programming: XP).
  Internet: http://jummp.wordpress.com/2011/04/20/desarrollo-de-software-programacion-extrema-extreme-programming-xp/
  Acceso: 05/08/2011
- XP on One Page
  Internet: http://xp123.com/articles/xp-on-one-page/
  Acceso: 05/08/2011
- Extreme Programming Roadmap
  Internet: http://c2.com/cgi/wiki?ExtremeProgramming
  Acceso:05/08/2011
- Una introducción al Extreming Programming
  Internet:http://www.dosideas.com/noticias/metodologias/822-una-introduccion-a-extreme-programming.html
  Acceso: 08/08/2011
- eXtreme Programming en ADWE Madrid
  Internet:http://www.jsanchez.es/extreme-programming-adwe-madrid/6480
  Acceso: 08/08/2011
- Extreme Programming:A gentle introduction
  Internet:http://www.extremeprogramming.org/
  Acceso:10/08/2011
- Programación extrema
  Internet:http://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema
  Acceso:10/08/2011
- Extreme programming:Origins
  Internet:http://en.wikipedia.org/wiki/Extreme_Programming#Origins
  Acceso:10/09/2011
- Extreme programming (Archivo de Power Point)
  Internet:http://www.slideshare.net/joaquin_win/extreme-programming-456979
  Acceso:11/09/2011

- A propósito de programación extrema XP (eXtreme Programming)
  Internet:http://www.monografias.com/trabajos51/programacion-extrema/programacion-extrema.shtml
  Acceso:12/09/2011
- PX(Programación Extrema)
  Internet: http://www.programacionextrema.org/
  Acceso: 12/09/2011
- What is Extreme Programming?
  Internet: http://xprogramming.com/book/whatisxp/
  Acceso: 12/09/2011
- Extreme Programming FAQ
  Internet: http://www.jera.com/techinfo/xpfaq.html
  Acceso: 12/09/2011
- Manifiesto ágil
  Internet: http://es.wikipedia.org/wiki/Manifiesto_%C3%A1gil
  Acceso: 14/09/2011
- The Four XP Values
  Internet:http://www.softwarereality.com/lifecycle/xp/four_values.jsp
  Acceso:14/09/2011
- Extreme programming
  Internet:http://en.wikipedia.org/wiki/Extreme_Programming#Origins
  Acceso:14/09/2011
- Características del lenguaje Java
  Internet: http://www.iec.csic.es/criptonomicon/java/quesjava.html
  Acceso: 13/09/2011
- I.3. CARACTERÍSTICAS DE JAVA
  Internet: http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_3.htm
  Acceso: 13/09/2011
- JAVA: Características del lenguaje
  Internet: http://tikal.cifn.unam.mx/~jsegura/LCGII/java3.htm
  Acceso:13/09/2011

- MANUAL DE JAVA
  Internet:http://www.webtaller.com/manual-java/caracteristicas-java.php
  Acceso: 13/09/2011
  Java (lenguaje de programación)
- Internet:http://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)
  Acceso: 14/09/2011
- Paradigma de programación
  Internet:http://es.wikipedia.org/wiki/Paradigma_de_programaci%C3%B3n
  Acceso: 14/09/2011
- Programación orientada a objetos
  Internet:http://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos
  Acceso: 14/09/2011
- UN POCO MAS DE JAVA: HISTORIA DE JAVA
  Internet: http://yosh35.tripod.com/id2.html
  Acceso: 15/09/2011
- Programación Orientada a Objetos
  Internet:http://www.monografias.com/trabajos/objetos/objetos.shtml
  Acceso: 15/09/2011
- Programación Orientada a Objetos – Eventos (I)
  Internet:http://thefricky.wordpress.com/2008/04/23/programacion-orientada-a-objetos-eventos-i/
  Acceso: 16/09/2011
- El lenguaje Java
  Internet: http://www.dcc.uchile.cl/~lmateu/Java/Apuntes/java.htm
  Acceso: 16/09/2011
- Eliminar Objetos no Utilizados
  Internet:http://artemisa.unicauca.edu.co/~dparedes/java/objetos/eliobjetos.html
  Acceso: 16/09/2011
- Encapsulamiento (P.O.O)
  Internet: http://damian02.wordpress.com/encapsulamiento-poo/
  Acceso: 20/09/2011

- Ocultación de datos
  Internet: http://es.kioskea.net/contents/poo/encapsul.php3
  Acceso: 20/09/2011
- MODULARIDAD
  Internet: http://labsys.frc.utn.edu.ar/aed-2005/Unidad%20II/Apunte/U2%20-%20Modularizaci%F3n.pdf
  Acceso: 20/09/2011
- POO
  Internet: http://www.elrinconcito.com/articulos/POO/POO.htm
  Acceso: 20/09/2011
- Por qué «software libre» es mejor que «código abierto»
  Internet: http://www.gnu.org/philosophy/free-software-for-freedom.es.html
  Acceso: 22/09/2011
- Software libre vs código abierto
  Internet: http://ideaschile.wordpress.com/2010/01/21/software-libre-vs-codigo-abierto/
  Acceso: 22/09/2011
- Breve historia del software libre
  Internet: http://innovacion.ticbeat.com/breve-historia-software-libre/
  Acceso: 22/09/2011
- ¿Qué es software libre?
  Internet: http://www.gnu.org/philosophy/free-sw.es.html
  Acceso: 22/09/2011
- Copyleft
  Internet: http://es.wikipedia.org/wiki/Copyleft
  Acceso: 22/09/2011

ANEXO A

ESCUELA FISCAL MIXTA

"TEODORO WOLF" Nº 72

SAN LORENZO- MANTA- MANABÌ

1. DATOS INFORMATIVOS.

- **NOMBRE DEL PLANTEL:** "TEODORO WOLF" Nº 72
- **CÓDIGO:** 13H02655
- **FECHA DE CREACIÓN**: 01-01-1948
- **DIRECTOR:** SR. PROF. JORGE ALBERTO MENDOZA SEGOVIA
- **SUPERVISOR: SR**. LCDO. HUMBOLDT DE LA TORRE ANDRADE
- **EMAIL:** escteodorowolf72@gmail.com
- **JORNADA:** MATUTINA
- **NIVEL:** EDUCACIÓN GENERAL BÁSICA
- **SOSTENIMIENTO**: FISCAL
- **REGIMEN**: COSTA
- **MODALIDAD**: PRESENCIAL
- **JURISDICCIÓN:** HISPANA
- **SEXO:** MIXTO
- **CLASE**: COMÚN
- **UBICACIÓN**: CALLE 10 DE AGOSTO Y MARGEN DEL RÍO
- **PARROQUIA:** SAN LORENZO
- **CANTÓN**: MANTA
- **PROVINCIA**: MANABÍ

VISTA EXTERIOR



VISTA INTERIOR



NÓMINA DEL PERSONAL DOCENTE





PROF. ÁLVAREZ DELGADO SANDER BOLÍVAR

LCDA. BAILÓN SALTOS CECILIA ALEXANDRA

LCDA. CEDEÑO RODRÍGUEZ ANA LORENA

LCDA. DE MERA SANTOS MARÍA ELENA

LCDO. GUADAMUD CANTOS JOSÉ GABRIEL

LCDA. MERO LÓPEZ GLORIA ESTRELLA

LCDO. RIVERA MACÍAS BORIS EDUARDO

ANEXO B

SISTEMA ACADEMICO NATLOG "S.A.N."

Manual para el Usuario.

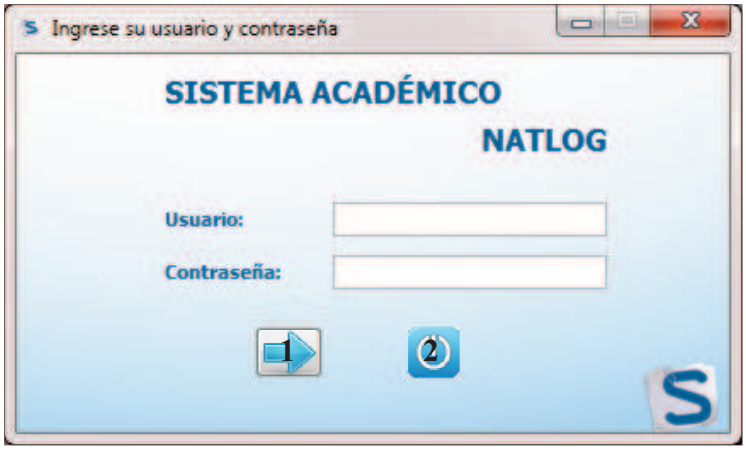


Tabla de Contenidos.

1. Ingresando al Sistema……..……1
2. Estudiantes……2
2.1. Ingresar Nuevo……...……2
2.2. Búsqueda General……..3
2.3. Buscar Código……4
3. Profesores……..6
3.1. Ingresar Nuevo……...6
3.2. Búsqueda General……..7
3.3. Buscar Cédula……7
4. Padres de Familia……..9
4.1. Ingresar Nuevo……...9
4.2. Búsqueda General……10
4.3. Buscar Código……..10
5. Materias……...14
5.1. Ingresar Nueva……14
5.2. Búsqueda General……15
5.3. Buscar Código……..15
6. Servicios……..18
6.1. Libreta de Calificaciones……18
6.2. Nuevo Año Escolar……..20
7. Impresiones……21
7.1. Libreta de Calificaciones……21
7.1.1. Curso……....21
7.1.2. Código……..23
7.2. Kardex……..23
7.2.1. Curso……23
7.2.2. Código……..24
7.3. Carnet……...24
7.3.1. Curso……24
7.3.2. Código……..25
7.4. Horarios……25
7.5. Certificado Anual……26
8. Estadísticas……..27
8.1. Horas por Profesor……...27
8.2. Notas por Curso……...28
8.3. Promedios Generales……...29
9. Respaldo BBD……30
9.1. Respaldo Actual……...30
10. Comandos Especiales y Recomendaciones……31

1. Ingresando al Sistema.

Estando en la pantalla "Escritorio" de su ordenador se hace doble clic en el icono del programa y a continuación aparecerá esta ventana:

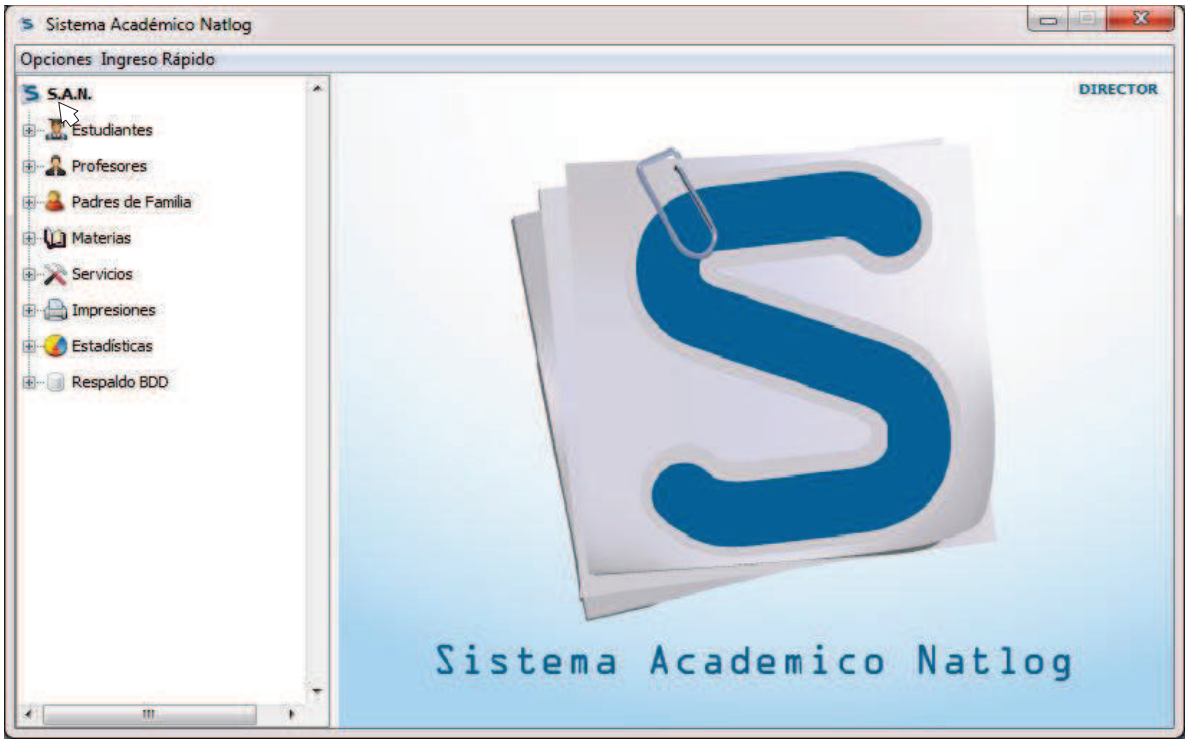


1.- Entrar.

2.- Salir.

Se ubica en las casillas lo requerido. El usuario y contraseña están registrados por default para el director (USUARIO: admin y CONTRASEÑA:12345) y este a su vez ingresara al sistema la cantidad de usuarios que se requieran.

Para que se desplieguen todas las opciones que ofrece el programa es necesario hacer doble clic sobre las siglas S.A.N.

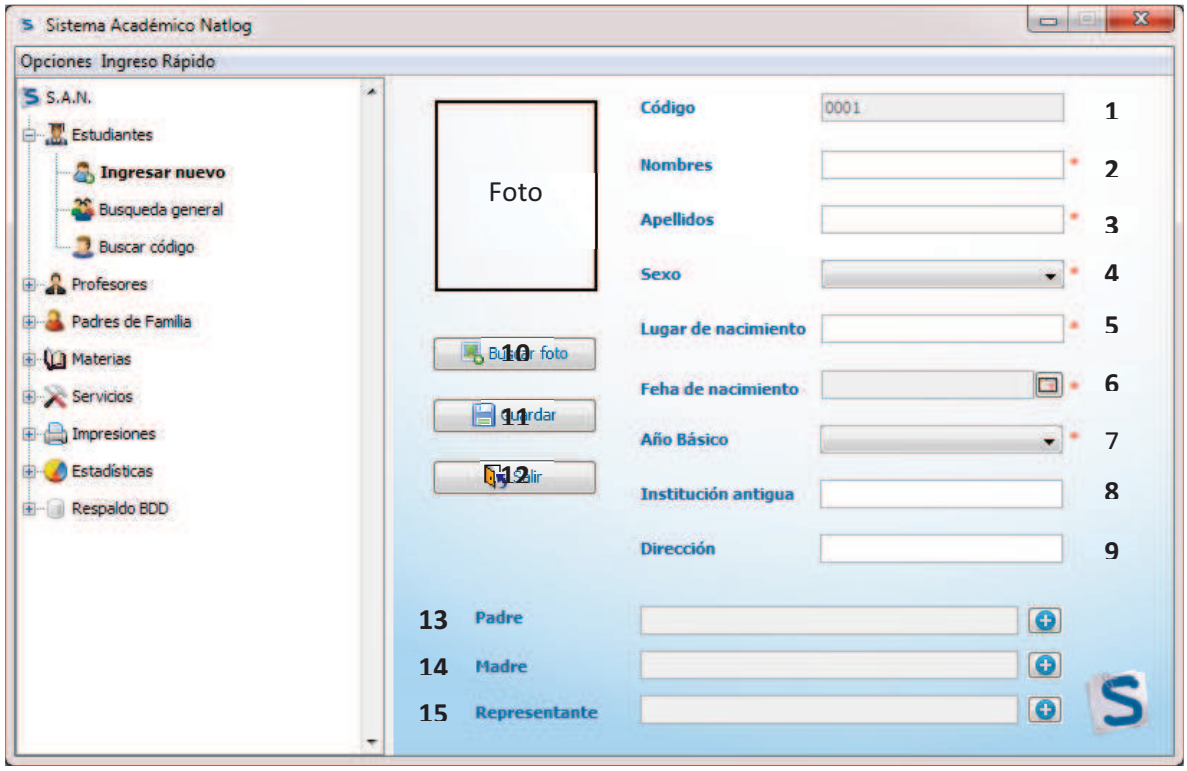


2. Estudiantes.

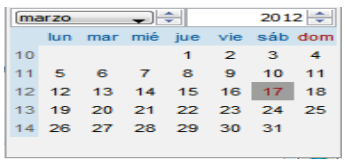
Se da clic en el signo más [+] a la izquierda de la palabra "Estudiantes" y se presentaran tres nuevas opciones.

- Ingresar Nuevo.
- Búsqueda General.
- Buscar Código.

2.1. Ingresar Nuevo.



1) Código: El sistema lo genera, es secuencial y no está sujeto a cambios.
2) Nombres: Se recomienda escribir los dos nombres del nuevo alumno.
3) Apellidos: Se aconseja escribir los dos apellidos del nuevo estudiante.
4) Sexo: Escoja entre "Masculino" y "Femenino".
5) Lugar de Nacimiento: Designe el lugar natal.
6) Fecha de nacimiento: Para mayor comodidad del usuario y facilitar el descifrado de los comandos del sistema se muestra un calendario para ingresar la fecha de nacimiento. De este se escoge el día del mes y se escribe el año.

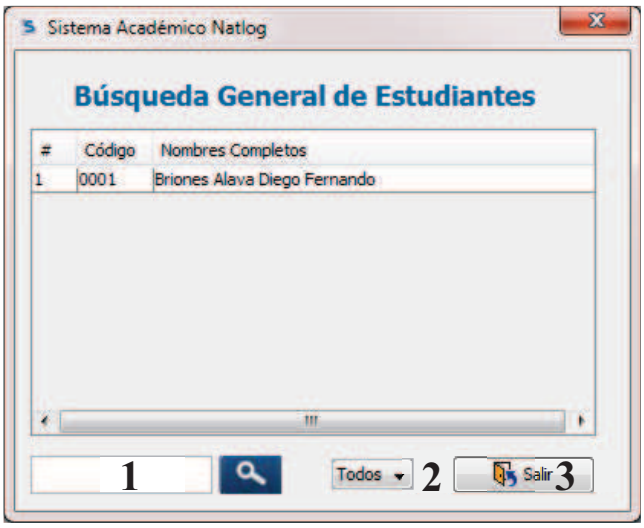


7) Año Básico: Se puede escoger desde el 1-7 año básico de acuerdo al estudiante.
8) Institución Antigua: Aquí se ubica el nombre de la institución previa a Teodoro Wolf N°72.
9) Dirección: Se ingresa la dirección actual del alumno.
10) Buscar Foto: Se abre una ventana para buscar una foto, que haya sido previamente almacenada en su ordenador, y ubicarla como distintivo visual del alumno. Máximo de 300kb(03 MB)
11) Guardar: Se hace clic en este botón una vez finalizado.
12) Salir: el ingreso de un nuevo estudiante.
13) Padre: Clic en el signo más [+] y aparecerá la base de datos con los nombres de los padres en el sistema, escoger el correspondiente al alumno.
14) Madre: Clic en el signo más [+] y se mostrara la base de datos de las mares en el sistema, se escoge el correspondiente.
15) Representante: Clic en el signo más [+] y se mostraran en conjunto las bases de datos tanto de padres, madres como representantes, debe escoger el correspondiente.

- Los campos con asteriscos (*) indican obligatoriedad.
- Los nombres de los padres, madres y representantes deben ser ingresados previamente para que estén disponibles en la base de datos. Véase el procedimiento para hacerlo en "Padres de Familia".

2.2. Búsqueda General.

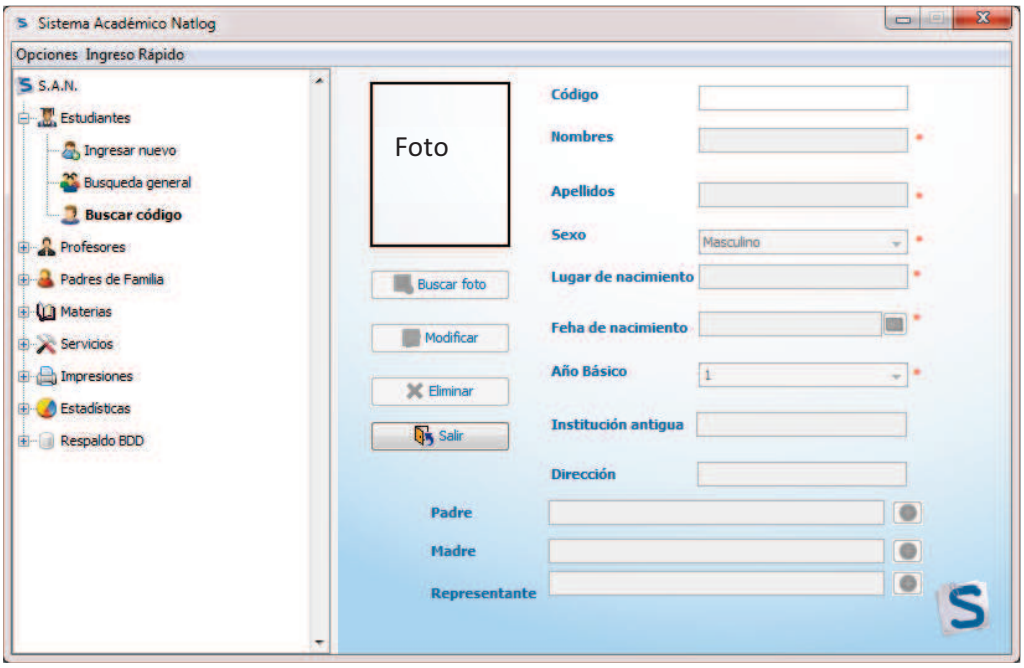
Se hace doble clic en la opción "Búsqueda General" y se muestra esta ventana:



1) Se ingresa el nombre, parcial o completo, del estudiante que se desee buscar.
2) El alcance de la búsqueda de acuerdo al curso. Puedes buscar en todos, 1, 2, 3, 4, 5, 6 y 7 año.
3) Botón para salir de esa ventana.

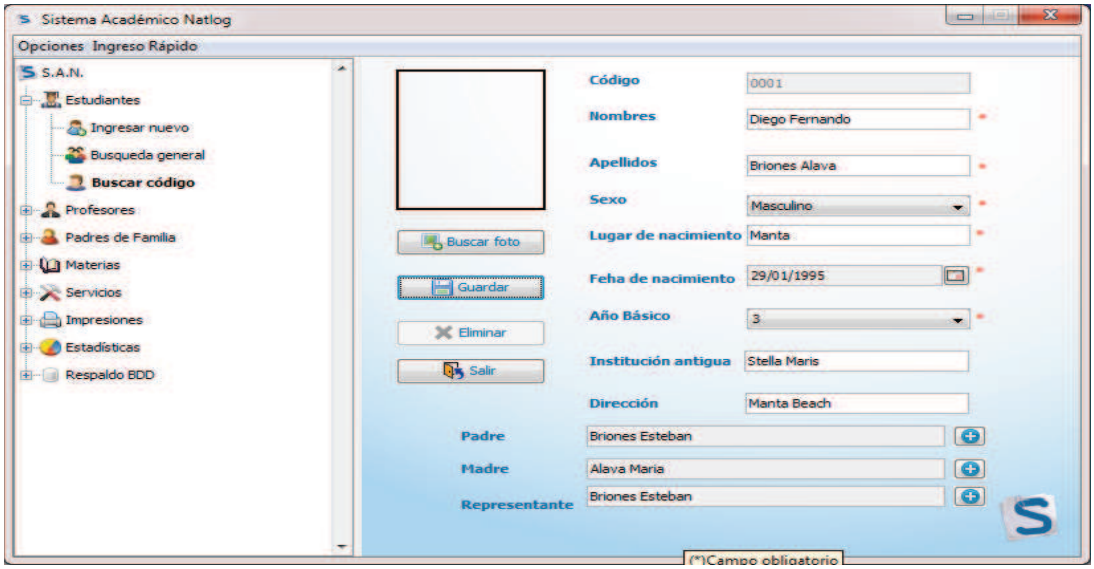
2.3. Buscar Código.

Para buscar un estudiante de acuerdo a su código se ubica en el campo que está a la derecha de la palabra "Código" y se escribe un número de 4 dígitos, que haya sido agregado al sistema previamente, para obtener los datos del alumno asignado a esta serie.

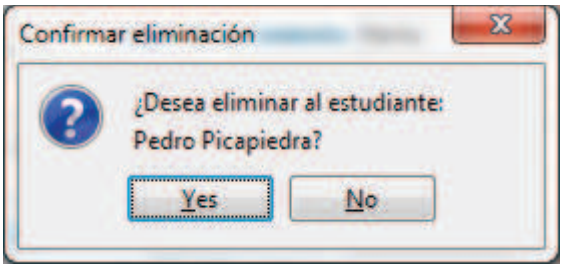


- En el instante que se abre la ventana, el único campo activado para escritura es el de Código. En el momento en que se escriba uno correcto se llenan todos los campos con los datos del alumno que posee este código.

- Para que se activen los otros campos y sean aptos para cambios se debe ingresar un código y hacer clic en el botón "Modificar". Una vez finalizados los cambios se hace clic en el botón "Guardar".



- Si se desea borrar a un alumno del sistema se ubica el código de este y se hace clic en el botón "Eliminar" luego aparece una ventana preguntando si está seguro de que desea eliminar a ese estudiante y usted escoge "Si" o "No" de acuerdo a lo requerido.



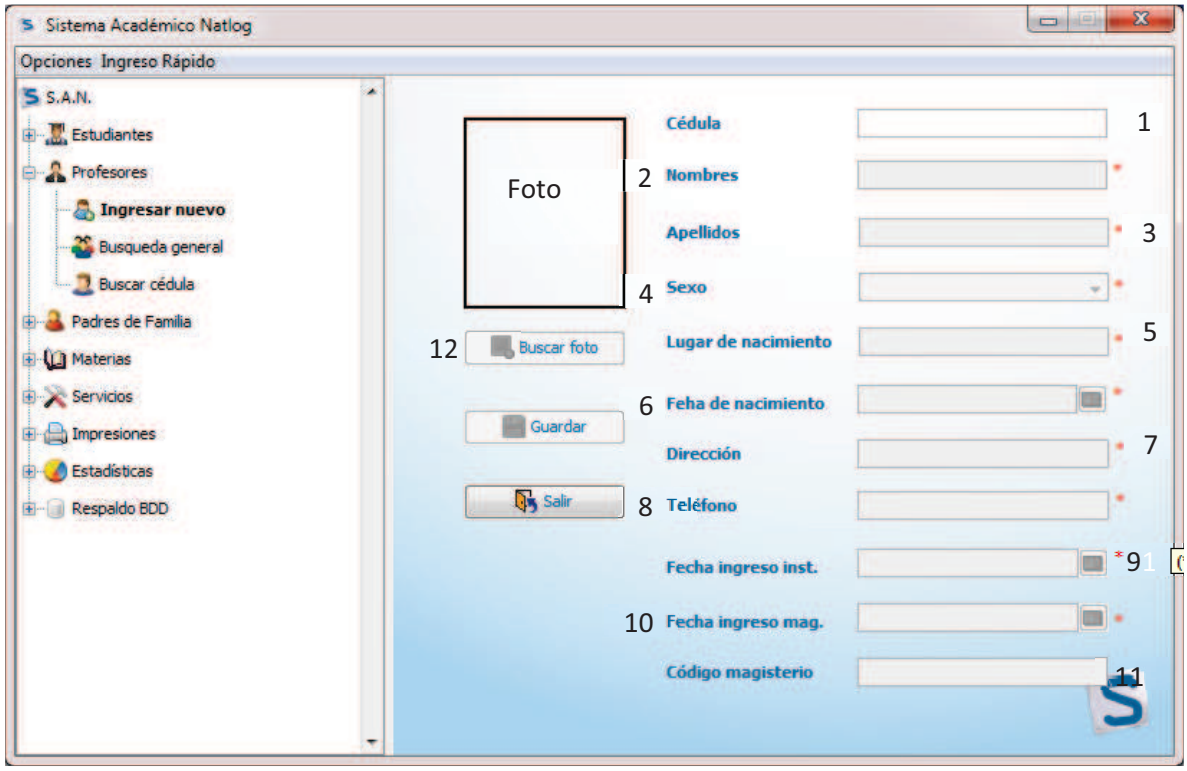
- Si decide que no desea realizar ninguna actividad relacionada con esta opción simplemente da clic en el botón "Salir".

3. Profesores.

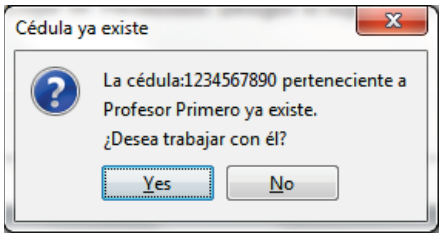
Se da clic en el signo más [+] a la izquierda de la palabra "Profesores" y se presentaran tres nuevas opciones.

- Ingresar Nuevo.
- Búsqueda General.
- Buscar Código.

3.1. Ingresar Nuevo.



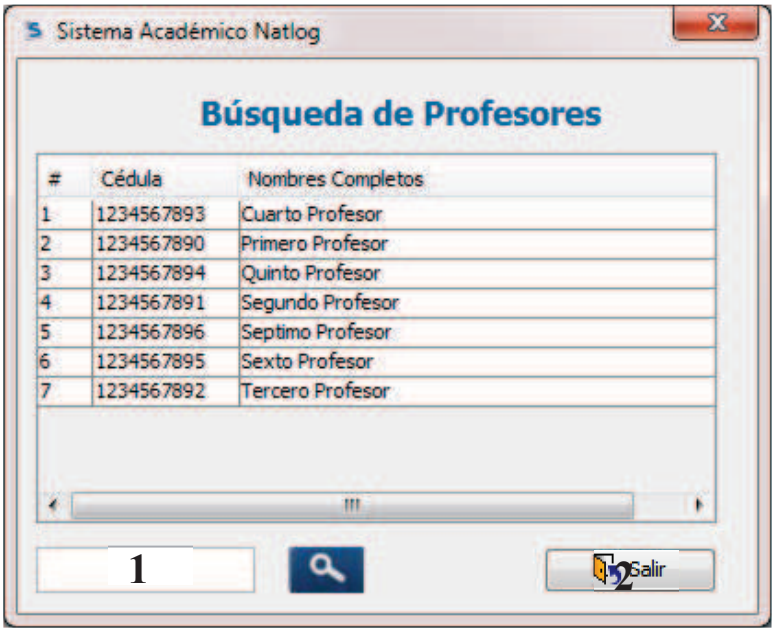
1) Cedula: Se introduce la cédula de identidad del profesor.
   - No se activaran los siguientes campos a menos que se introduzca una cedula con los 10 números correspondientes.
   - Si se intenta ingresar una cedula que ya existe en la base de datos se mostrará la siguiente ventana.



2) Nombres: Se recomienda escribir los dos nombres.
3) Apellidos: Se aconseja escribir los dos apellidos del nuevo docente.
4) Sexo: Escoja entre "Masculino" y "Femenino".
5) Lugar de Nacimiento: Designe el lugar natal.
6) Fecha de nacimiento: Para mayor comodidad del usuario y facilitar el descifrado de los comandos en el sistema se muestra un calendario para ingresar la fecha de nacimiento. De este se escoge el día del mes y se escribe el año.
7) Dirección: Se ingresa la dirección actual del alumno.
8) Teléfono: Se ingresa un número al cual se pueda localizar al profesor cuando sea necesario.
9) Fecha ingreso inst.: También mediante un calendario, se ubica la fecha en la que el profesor inicia sus labores en la institución.
10) Fecha ingreso mag.: Mediante el calendario se ubica la fecha de afiliación al Magisterio.
11) Código magisterio: Al momento de la afiliación a los profesores se les entrega un código del Magisterio y este debe ser ingresado en este campo.
12) La foto se clic en Botón Buscar y se busca la foto que desea colocar (El máximo tamaño de la foto es de 300kb o du equivalencia de 0.3MB)

3.2. Búsqueda General.

Se hace doble clic en la opción "Búsqueda General" y se muestra esta ventana

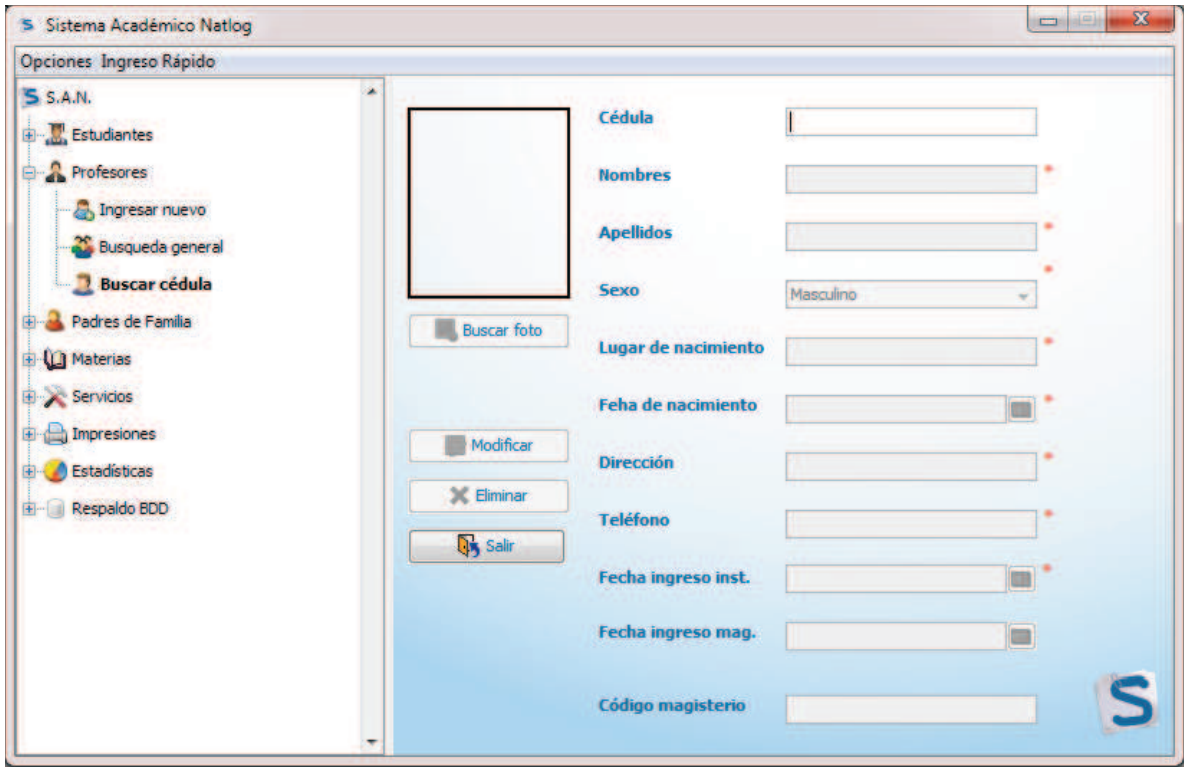


1) Se ingresa el nombre, parcial o completo, del profesor que se desee buscar.
2) Botón para salir de esa ventana.

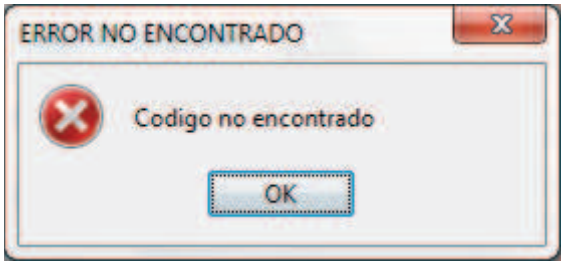
3.3. Buscar Cédula.

Con esta opción se busca a los profesores a través de su cédula de identidad.

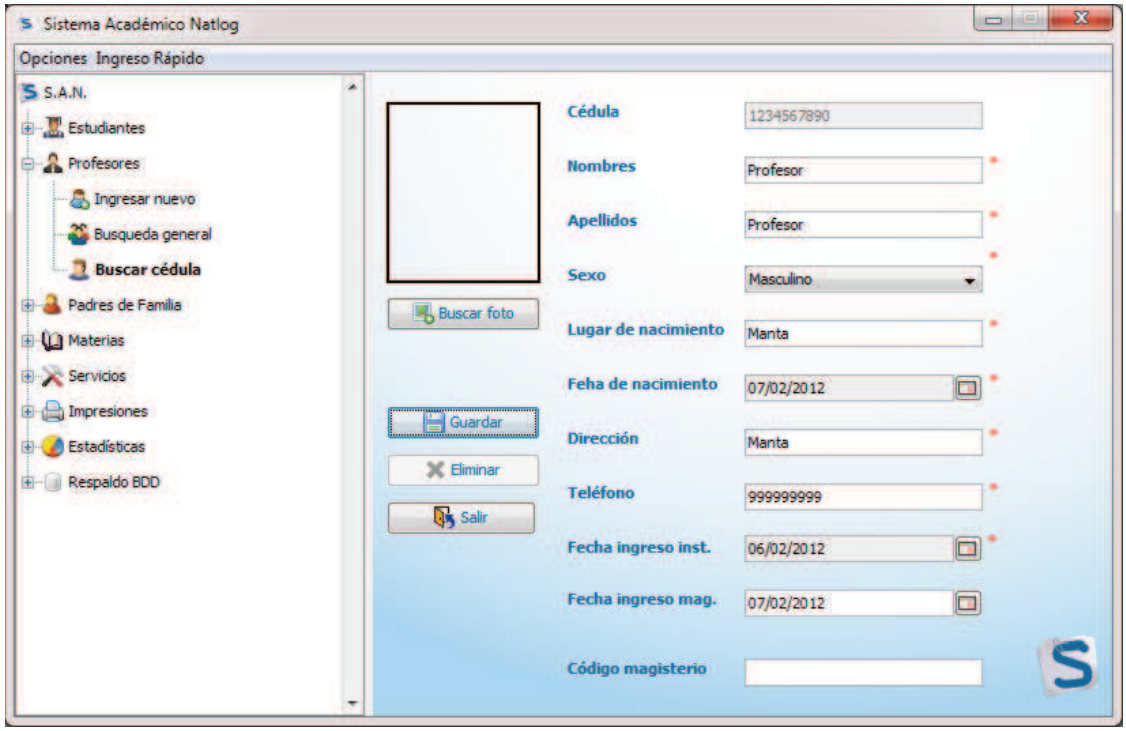
Se da doble clic en "Profesores" y luego doble clic en "Buscar cédula" y se mostrara lo siguiente:



- En el instante que se abre la ventana, el único campo activado para escritura es el de "Cédula". En el momento en que se escriba una correcta se llenan todos los campos con los datos del profesor que posee este cedulado.
- En caso de que la cedula no exista se mostrara el siguiente error.



Para que se activen los otros campos y sean aptos para cambios se debe ingresar un número de cédula y hacer clic en el botón "Modificar". Una vez finalizados los cambios se hace clic en el botón "Guardar".



- Si se desea borrar a un profesor del sistema se ubica el código de este y se hace clic en el botón "Eliminar" luego aparece una ventana preguntando si está seguro de que desea eliminar a esta persona y usted escoge "Si" o "No" de acuerdo a lo requerido.
- Los profesores no pueden borrar a otros profesores, esto es algo que solo puede realizar el director,
- Si decide que no desea realizar ninguna actividad relacionada con esta opción simplemente da clic en el botón "Salir".

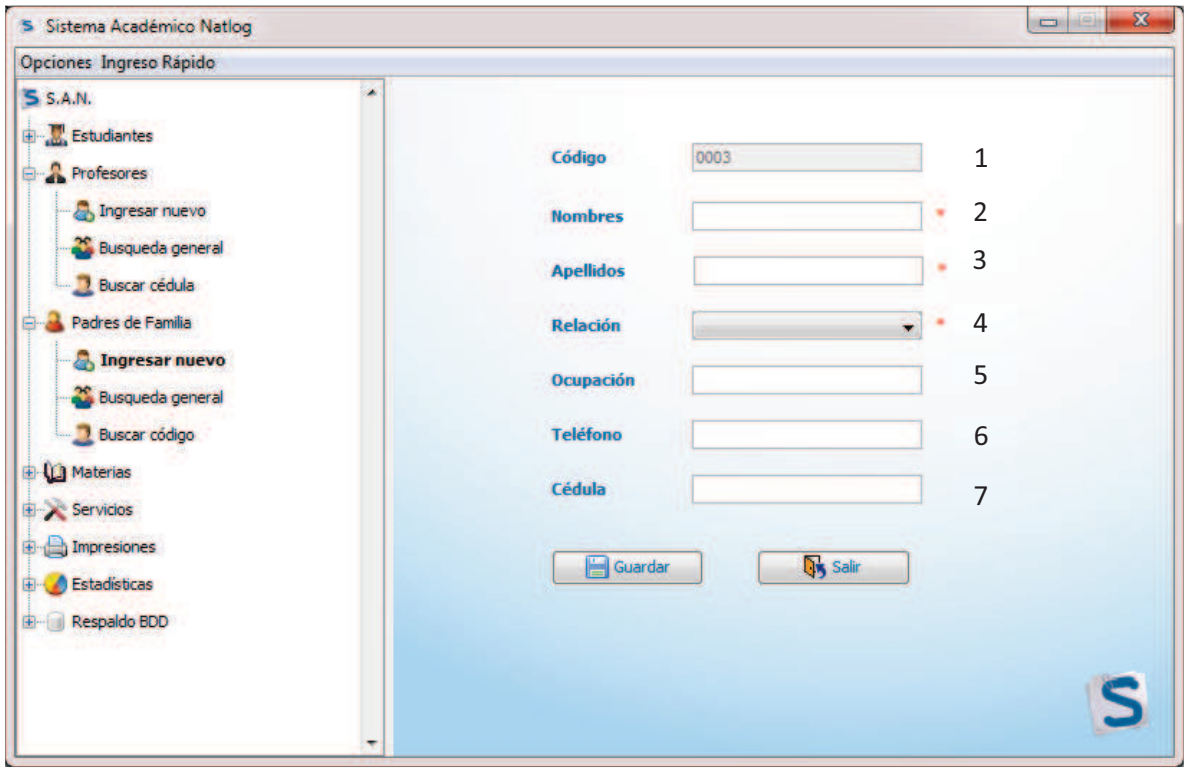
4. Padres de Familia.

Se da clic en el signo más [+] a la izquierda de las palabras "Padres de Familia" y se presentaran tres nuevas opciones.

- Ingresar Nuevo.
- Búsqueda General.
- Buscar Código.

4.1. Ingresar Nuevo.

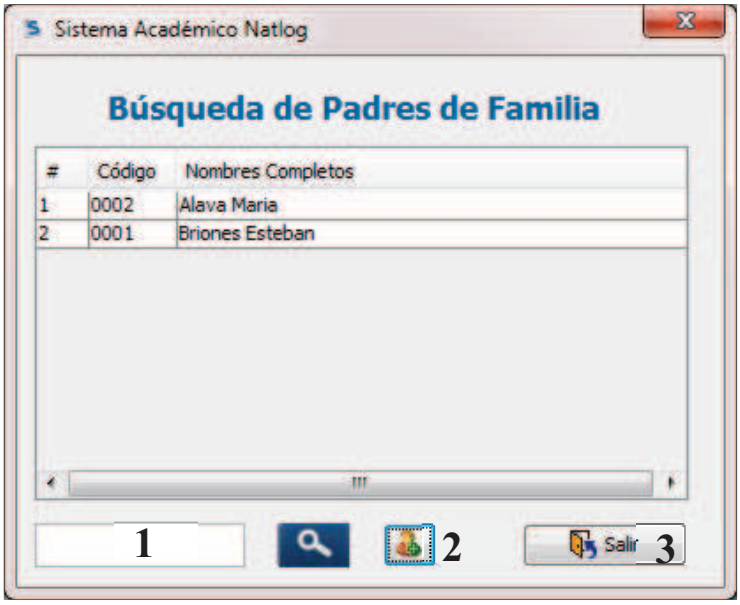
Opción que registra un nuevo padre de familia en el sistema.



1) Código: El sistema lo genera, es secuencial y no está sujeto a cambios.
2) Nombres: Se recomienda escribir los dos nombres del padre de familia.
3) Apellidos: Se aconseja escribir los dos apellidos.
4) Relación: Escoja entre "Padre", "Madre" o "Representante.
5) Ocupación: Ingrese en que ocupa la mayor parte de su tiempo el padre de familia, o bien especifique en que trabaja.
6) Teléfono: Se ingresa un número al cual se pueda localizar al padre de familia cuando sea necesario.
7) Cédula: Se ingresan los diez dígitos correspondientes a la cedula del padre de familia.

4.2. Búsqueda General.

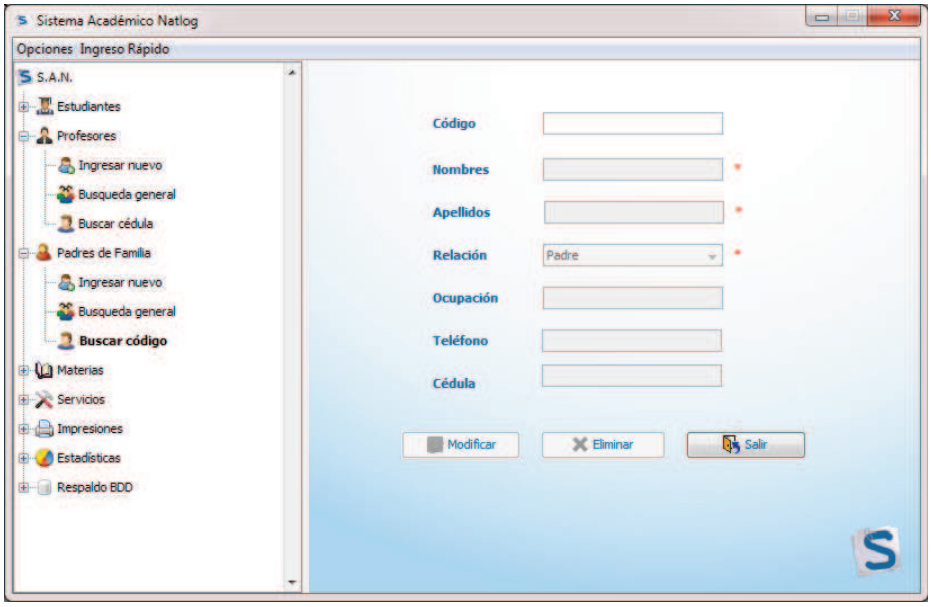
Se hace doble clic en la opción "Búsqueda General" y se muestra esta ventana



1) Se ingresa el nombre, parcial o completo, del padre de familia que se desee buscar.
2) Acceso directo para ingresar un nuevo padre en caso de que no esté en el sistema el padre solicitado.
3) Botón para salir de esa ventana.

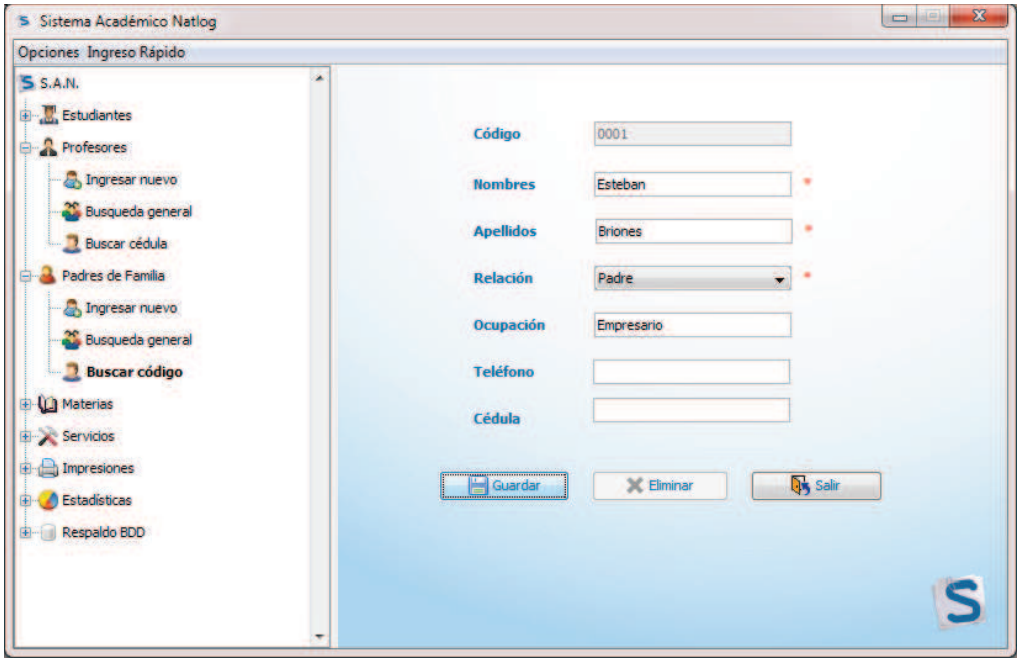
4.3. Buscar Código.

Para buscar un padre de familia de acuerdo a su código, se ubica en el campo que está a la derecha de la palabra "Código" y se escribe un número de 4 dígitos, que haya sido agregado al sistema previamente, para obtener los datos del representante asignado a esta serie.



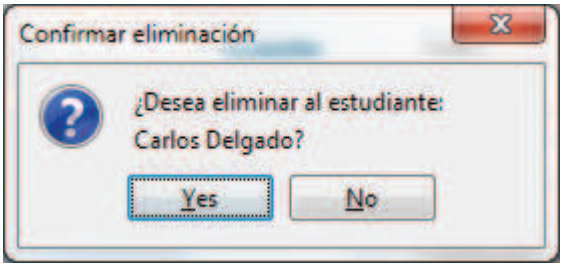
- En el instante que se abre la ventana, el único campo activado para escritura es el de Código. En el momento en que se escriba uno correcto se llenan todos los campos con los datos del padre de familia que posee este código.

- Para que se activen los otros campos y sean aptos para cambios se debe ingresar un código y hacer clic en el botón "Modificar". Una vez finalizados los cambios se hace clic en el botón "Guardar".



- Si se desea borrar a un padre de familia del sistema se ubica el código de este y se hace clic en el botón "Eliminar" luego aparece una ventana preguntando si está

seguro de que desea eliminar a esa persona del sistema y usted escoge "Si" o "No" de acuerdo a lo requerido.

- 

- Si decide que no desea realizar ninguna actividad relacionada con esta opción simplemente da clic en el botón "Salir".

5. Materias.

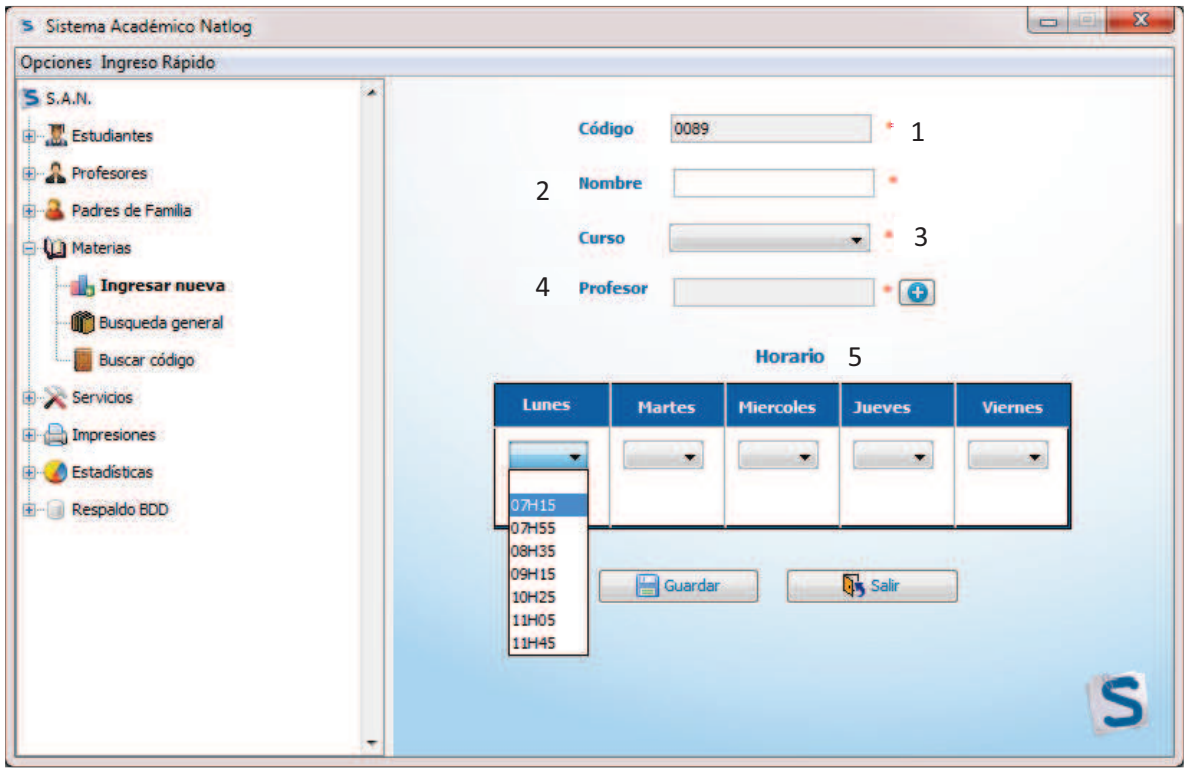
Se da clic en el signo más [+] a la izquierda de la palabra "Materias" y se presentaran tres nuevas opciones.

- Ingresar Nueva.
- Búsqueda General.
- Buscar Código.

5.1. Ingresar Nueva.

Opción que agrega una materia nueva a la base de datos.

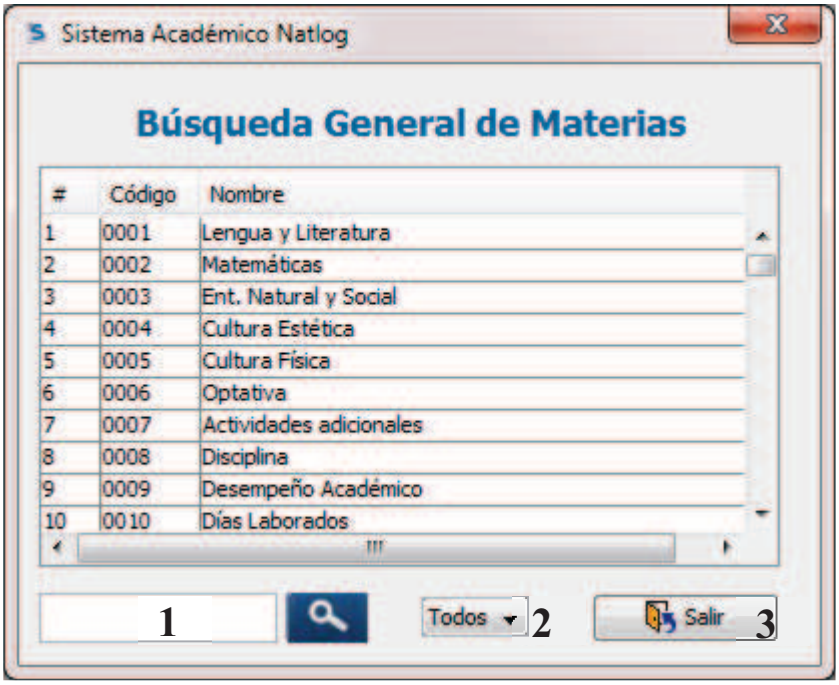
Doble clic en "Materias" Y luego en "Ingresar nueva" y se muestra esto:



1) Código: El sistema lo genera, es secuencial y no está sujeto a cambios.
2) Nombre: Aquí se escribe el nombre de la asignatura.
3) Curso: Se escoge el curso al que se le impartirá la materia. Se puede seleccionar desde el curso 1 hasta el 7.
4) Profesor: Se hace clic en el signo más [+] y se escoge uno de los profesores previamente ingresados (véase 3. Profesores – 3.1 Ingresar nuevo).
5) Horario: Se seleccionan las horas y los días en las que se darán las clases de esta nueva materia.

5.2. Búsqueda General.

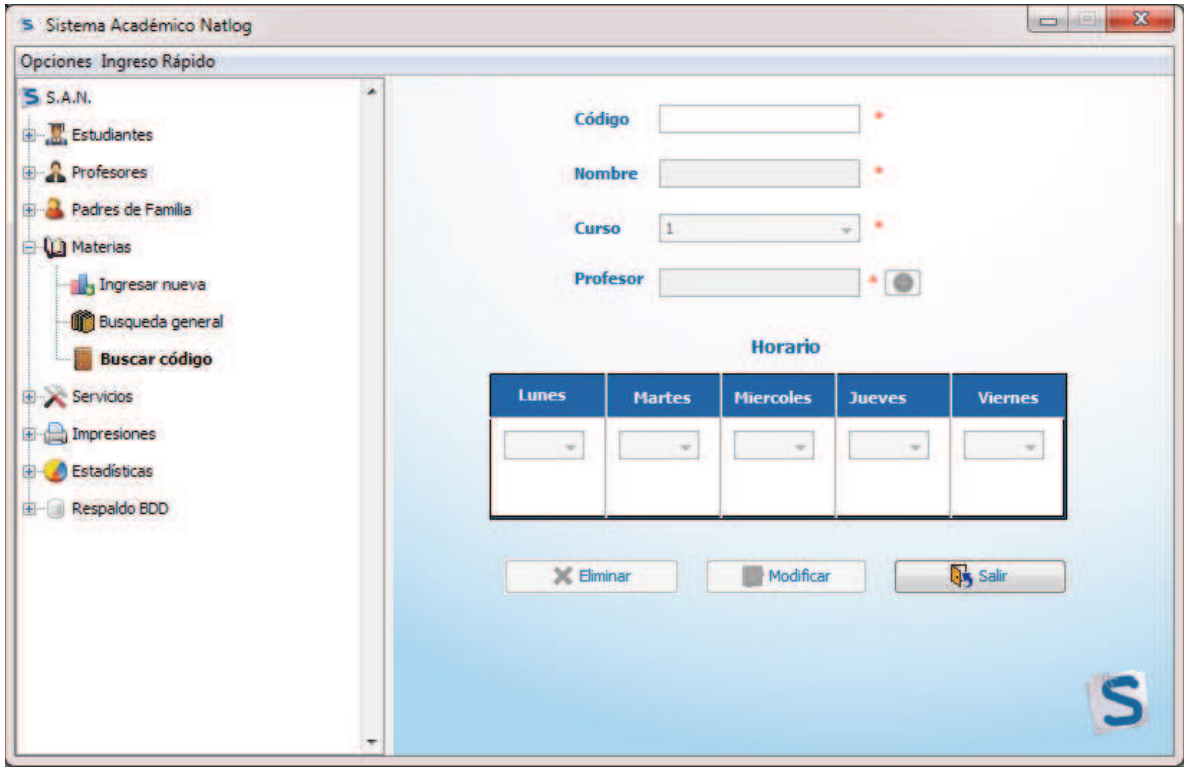
Dando doble clic en la opción "Búsqueda general" aparece esta ventana:



1) Se ingresa el nombre, parcial o completo, de la materia que se desee buscar.
2) El alcance de la búsqueda de acuerdo al curso. Puedes buscar en todos, 1, 2, 3, 4, 5, 6 o 7.
3) Botón para salir de esa ventana.

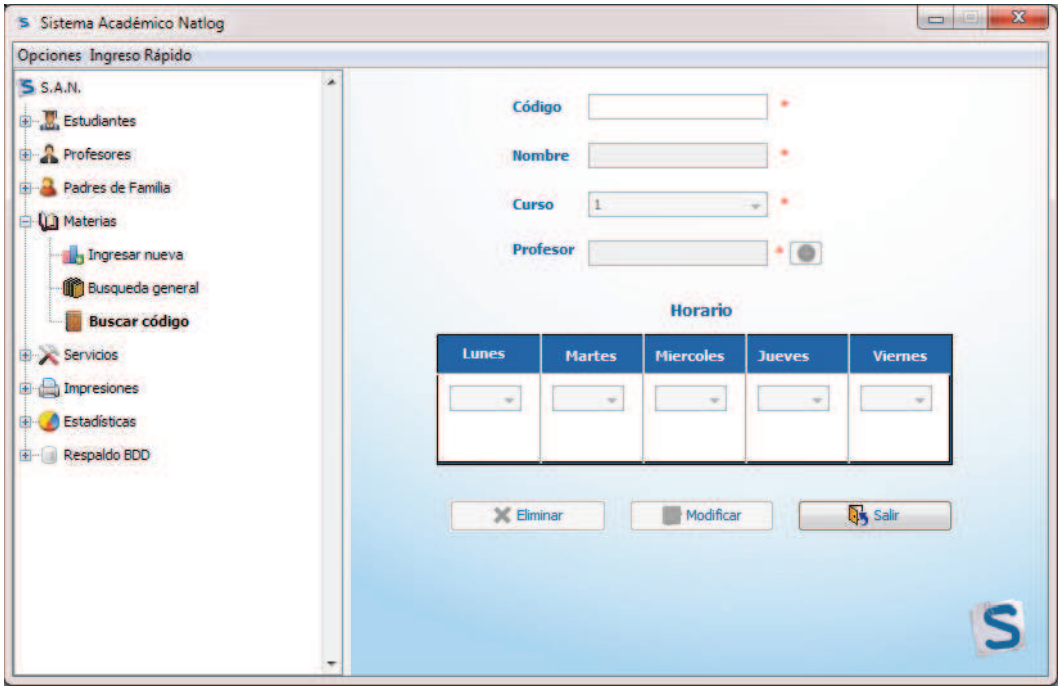
5.3. Buscar Código.

Para buscar una materia de acuerdo a su código, se ubica en el campo que está a la derecha de la palabra "Código" y se escribe un número de 4 dígitos perteneciente a una materia que haya sido agregada al sistema previamente, para obtener los datos de esta.



- En el instante que se abre la ventana, el único campo activado para escritura es el de Código. En el momento en que se escriba uno correcto se llenan todos los campos con los datos de la asignatura.

- Para que se activen los otros campos y sean aptos para cambios se debe ingresar un código y hacer clic en el botón "Modificar". Una vez finalizados los cambios se hace clic en el botón "Guardar".



- Si se desea borrar una materia del sistema se ubica el código de esta y se hace clic en el botón "Eliminar" luego aparece una ventana preguntando si está seguro de que desea eliminar esta materia y usted escoge "Si" o "No" de acuerdo a lo requerido.
- Si decide que no desea realizar ninguna actividad relacionada con esta opción simplemente da clic en el botón "Salir".

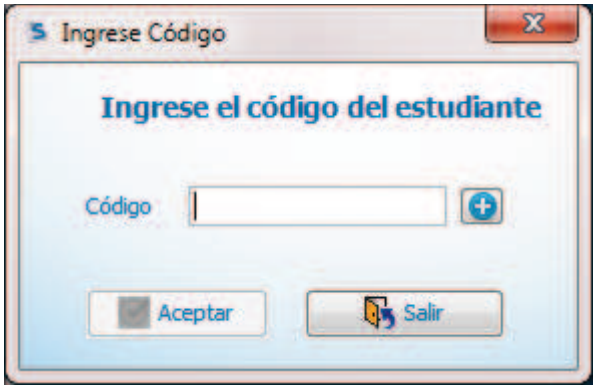
6. Servicios.

Se da clic en el signo más [+] a la izquierda de la palabra "Servicios" y se presentaran dos nuevas opciones.

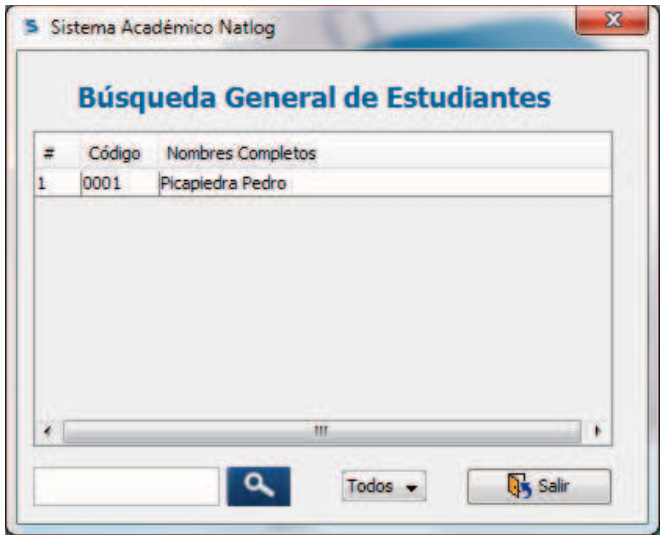
- Libreta de Calificaciones.
- Nuevo Año Escolar.

6.1. Libreta de Calificaciones.

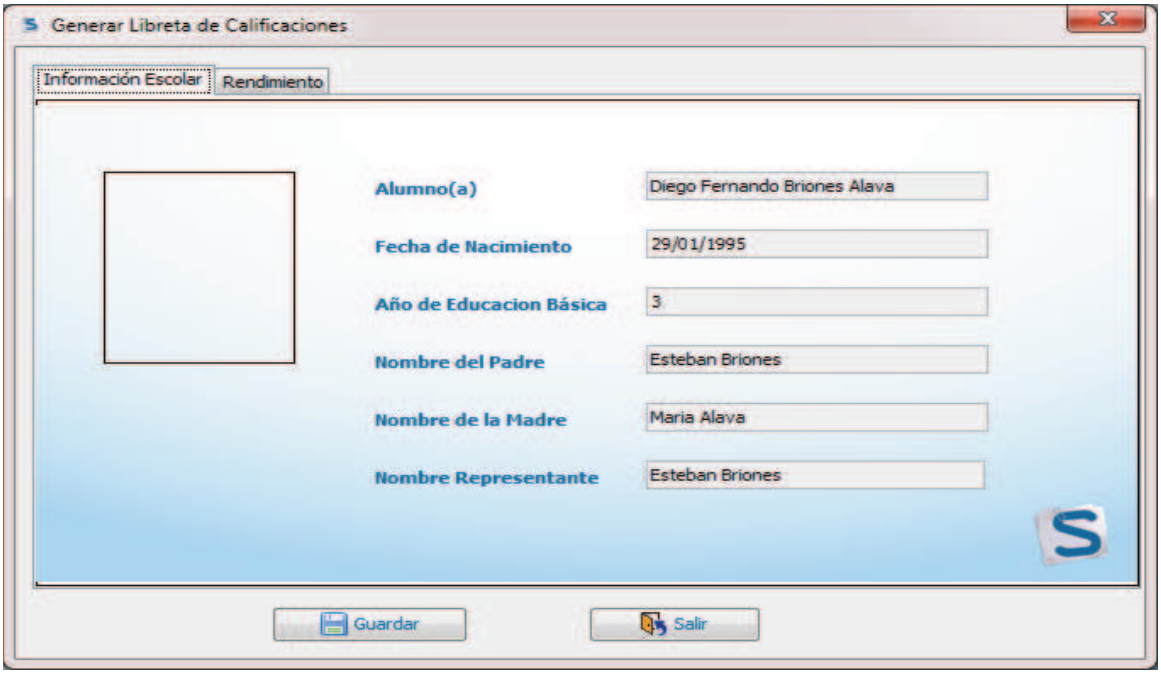
Clic en "Libreta de calificaciones" y aparecerá esta venta:



En caso de saberse el codigo del estudiante se lo ingresa directamente, sino se hace clic en el más [+] a la derecha del campo de escritura y se escoge uno de la base de estudiantes (se da clic sobre el que se quiere) y se hace clic en el boton "Aceptar".



Una vez escogido uno aparecera una ventana con la informacion del estudiante para confirmar que este es el alumno requerido:



A continuación se hace clic en la pestaña que dice "Rendimiento" y se mostraran las notas del estudiante:

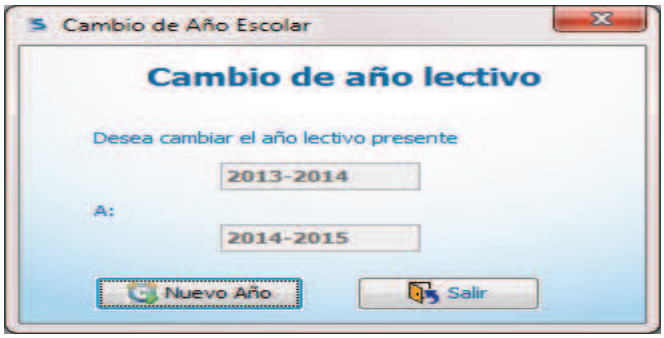


- Los campos seran llenados a decisión de los profesores y los promedios son sacados automaticamente por el programa.
- Solo se deben emplear números en los campos de esta ventana.
- Una vez finalizados los cambios se da clic al boton "Guardar".
- En caso de no querer realizar cambios solo haga clic en el boton "Salir".

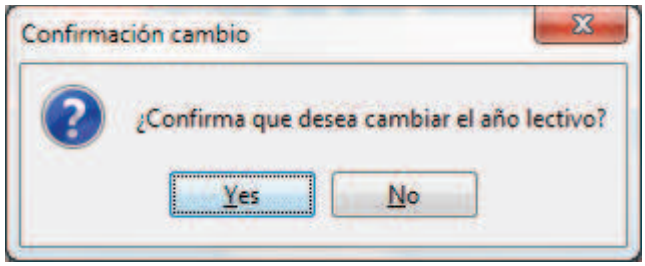
6.2. Nuevo Año Escolar.

Esta opción se utiliza cuando se concluye el año lectivo y se desea comenzar con el siguiente.

Clic en "Nuevo año escolar" y saldrá esta ventana:



- Clic en "Nuevo Año" y aparecerá un aviso pidiendo confirmar si está seguro que desea cambiar al siguiente año lectivo, en caso de estarlo haga clic en "Si" y si no lo está de clic en "No".



- Si no desea cambiar el año y por error ingresó a esta opción simplemente haga clic en "Salir".

7. Impresiones.

Esta opción permite imprimir diferentes documentos que podría requerir la institución.

Al hacer clic en este menú se despliegan los siguientes ítems:

- Libreta de calificaciones.
- Kardex.
- Carnet.
- Horarios.
- Certificado Anual.

7.1. Libreta de Calificaciones.

Específica para imprimir un boletín de notas.

Clic en "Libreta de calificaciones" y se presentan las posibilidades de imprimir la libreta de un estudiante según su:

- Curso.
- Código.

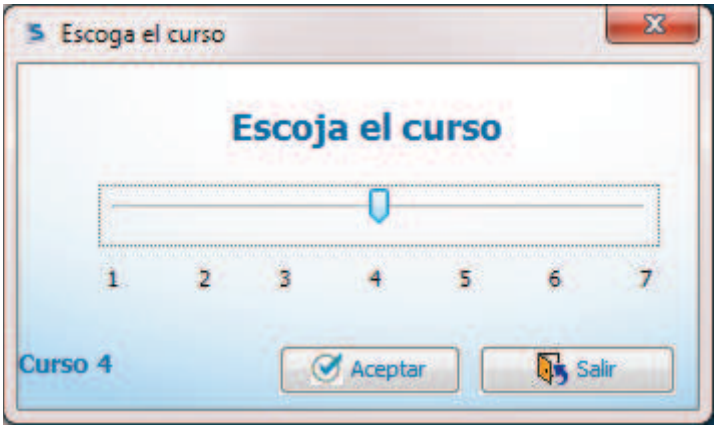
7.1.1. Curso.

Si se escoge la opción "Curso" aparecerá la ventana

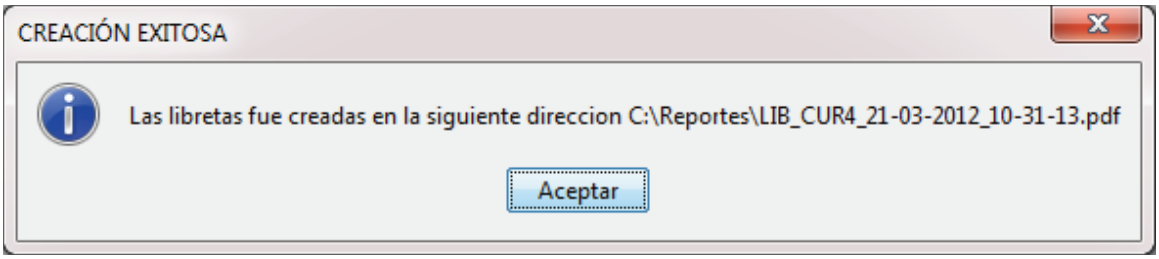


En la cual se ubica el puntero en la línea y sobre el número del curso que se desea buscar, como el ejemplo en el que se desean imprimir las libretas del curso 4 y se da clic hasta que el indicador este sobre este número mostrándose en la esquina inferior izquierda el curso escogido.

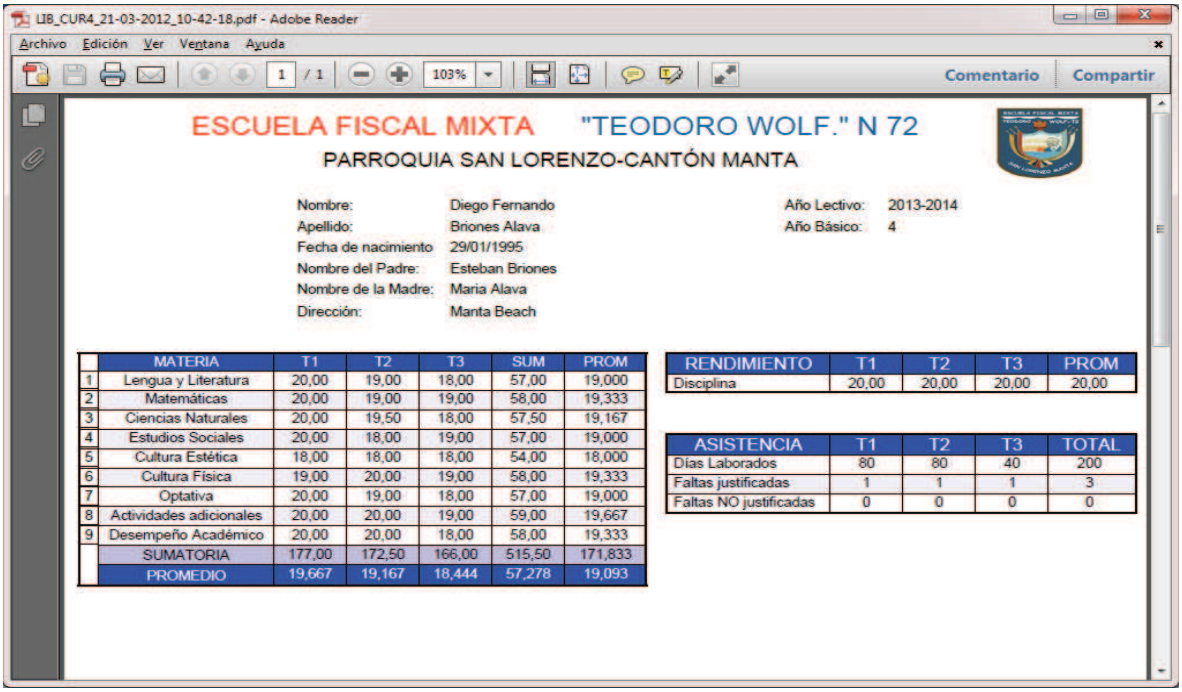
En caso de no desear realizar ninguna impresión se hace clic en el botón "Salir".



A continuación se da clic en el botón "Aceptar" y se desplegara este mensaje:



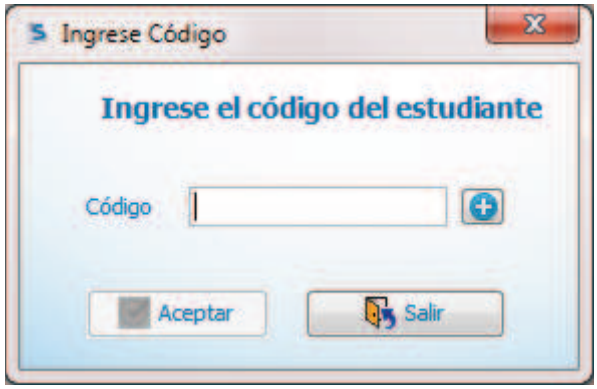
Indicando la dirección en donde fueron guardadas las libretas de ese curso. Se da clic en "Aceptar" y se mostrara el archivo PDF con las libretas de calificaciones. Un parecido a este:



Ahora puede imprimir las libretas de manera sencilla con la opción de impresión que ofrecen todos los archivos PDF.

7.1.2. Código.

El proceso es idéntico al de Código, solo difiere en el hecho de que ahora se buscan las libretas a imprimir de acuerdo al código y a través de esta ventana:



se ingresa el código del estudiante en caso de saberlo o si no se da clic en el signo más [+] a la derecha del campo y se presentara la ventana de "Búsqueda General de Estudiantes" (véase 2. Estudiantes – 2.2 Búsqueda general) se escoge uno haciendo doble clic sobre el nombre y cuando se regrese a la ventana anterior se hace clic en "Aceptar".

Y aparecerá primero el mensaje indicando donde está ha guardado el archivo abriéndose, a continuación, el documento PDF siendo apta para imprimirse la libreta.

En caso de no querer realizar la impresión se hace clic en el botón "Salir".

7.2. Kardex.

Específica para imprimir un historial de notas.

Clic en "Kardex" y se presentan las posibilidades de imprimir el kardex de un estudiante según su:

- Curso.
- Código.

7.2.1. Curso.

El sistema en esta opción se maneja de una forma parecida a 7.Impresiones – 7.1. Libreta de Calificaciones – 7.1.1. Curso. Así que sugerimos seguir los mismos pasos. La única

diferencia significativa es que el archivo PDF quedaría de la siguiente forma:



	MATERIA	TRIMESTRE I	TRIMESTRE II	TRIMESTRE III	SUMATORIA	PROM	CURSO	AÑO LECTIVO
1	Lenguaje y Literatura	19.547			19.547	19.547	5	2012-2013
2	Matemáticas	19.000			19.000	19.000	5	2012-2013
3	Ciencias Naturales	18.000			18.000	18.000	5	2012-2013
4	Estudios Sociales	14.000			14.000	14.000	5	2012-2013
5	Cultura Estética	18.000			18.000	18.000	5	2012-2013
6	Cultura Física	16.000			16.000	16.000	5	2012-2013
	SUMATORIA	104.547			104.547	104.547		
	PROMEDIO	17.424			17.424	17.424		

7.2.2. Código.

El proceso a seguir es similar a 7.Impresiones – 7.1. Libreta de Calificaciones – 7.1.2. Código y se mostrará el kardex, a manera de archivo PDF, del estudiante seleccionado.

7.3. Carnet.

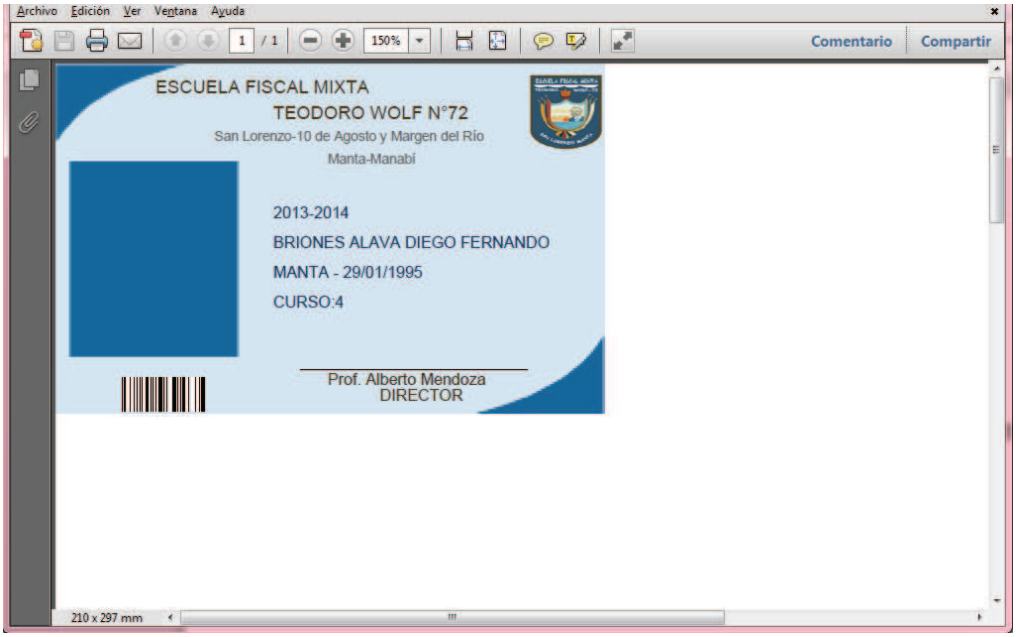
Específica para imprimir una identificación estudiantil.

Clic en "Carnet" y se presentan las posibilidades de imprimir el kardex de un estudiante según su:

- Curso.
- Código.

7.3.1. Curso.

Se siguen los mismos pasos que en 7.Impresiones – 7.1. Libreta de Calificaciones – 7.1.1. Curso. La diferencia está en que la ventana con el PDF que contiene los carnets del curso escogido será algo así:



Y se mostrarán los carnets del curso seleccionado.

- ✓ En el caso de no querer realizar nadase hace clic en el botón "Salir" en la ventana inicial de esta opción.

7.3.2. Código.

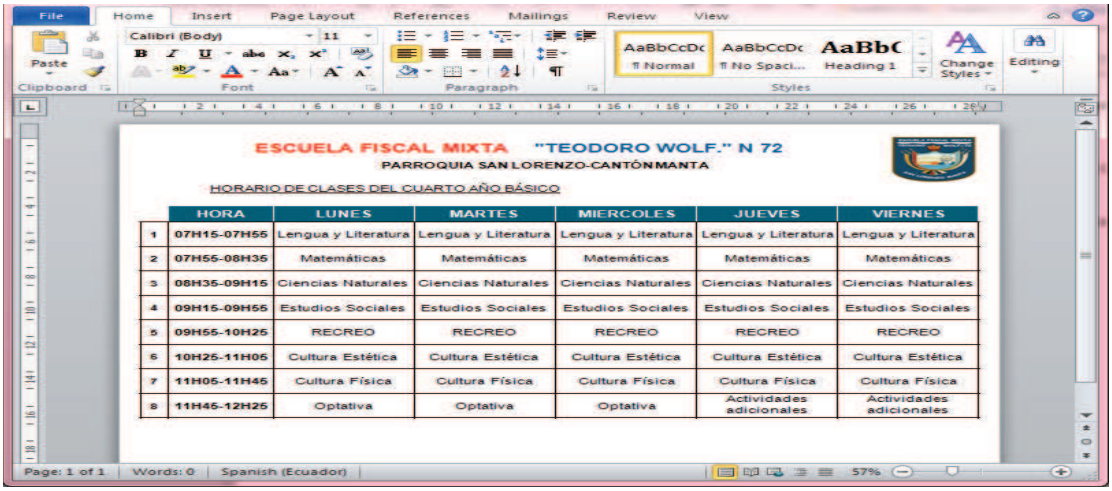
Los pasos son similares a 7.Impresiones – 7.1. Libreta de Calificaciones – 7.1.2. Código, se recomienda seguirlos para que se muestre el carnet del alumno requerido.

- **Los documentos PDF no son alterables, es decir, no se le pueden realizar cambios.**

7.4. Horarios.

Doble clic en "Impresiones" y luego clic en "Horarios" a continuación se siguen los mismos pasos que en 7.Impresiones – 7.1. Libreta de Calificaciones – 7.1.1. Curso.

Una vez realizados los pasos se abrirá un documento en Word, como este:



ESCUELA FISCAL MIXTA "TEODORO WOLF." N 72

PARROQUIA SAN LORENZO-CANTÓN MANTA

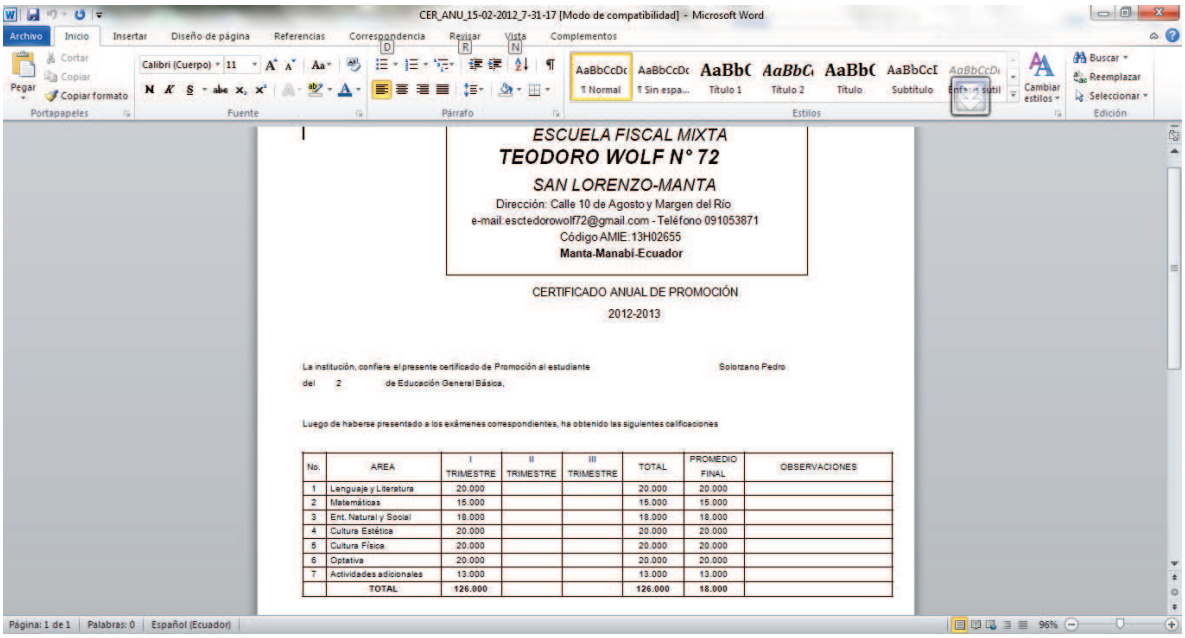
HORARIO DE CLASES DEL CUARTO AÑO BÁSICO

	HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
1	07H15-07H55	Lengua y Literatura	Lengua y Literatura	Lengua y Literatura	Lengua y Literatura	Lengua y Literatura
2	07H55-08H35	Matemáticas	Matemáticas	Matemáticas	Matemáticas	Matemáticas
3	08H35-09H15	Ciencias Naturales	Ciencias Naturales	Ciencias Naturales	Ciencias Naturales	Ciencias Naturales
4	09H15-09H55	Estudios Sociales	Estudios Sociales	Estudios Sociales	Estudios Sociales	Estudios Sociales
5	09H55-10H25	RECREO	RECREO	RECREO	RECREO	RECREO
6	10H25-11H05	Cultura Estética	Cultura Estética	Cultura Estética	Cultura Estética	Cultura Estética
7	11H05-11H45	Cultura Física	Cultura Física	Cultura Física	Cultura Física	Cultura Física
8	11H45-12H25	Optativa	Optativa	Optativa	Actividades adicionales	Actividades adicionales

- A este documento se le pueden realizar cambios antes de imprimirlo.
- En caso de no querer imprimir el horario solo se hace clic en el botón "Salir de la ventana inicial de esta opción.

7.5. Certificado Anual.

Se hace doble clic en "Impresiones" y luego clic en "Certificado anual", después de esto se realiza el mismo proceso que en 7.Impresiones – 7.1. Libreta de Calificaciones – 7.1.2. Código y al finalizarlos se mostrará el certificado del estudiante que es más o menos así:



ESCUELA FISCAL MIXTA
TEODORO WOLF N° 72
SAN LORENZO-MANTA
Dirección: Calle 10 de Agosto y Margen del Río
e-mail:esctedorowolf72@gmail.com - Teléfono 091053871
Código AMIE:13H02655
Manta-Manabí-Ecuador

CERTIFICADO ANUAL DE PROMOCIÓN
2012-2013

La institución, confiere el presente certificado de Promoción al estudiante Solorzano Pedro
del 2 de Educación General Básica,

Luego de haberse presentado a los exámenes correspondientes, ha obtenido las siguientes calificaciones

No.	AREA	I TRIMESTRE	II TRIMESTRE	III TRIMESTRE	TOTAL	PROMEDIO FINAL	OBSERVACIONES
1	Lenguaje y Literatura	20.000			20.000	20.000	
2	Matemáticas	15.000			15.000	15.000	
3	Ent. Natural y Social	18.000			18.000	18.000	
4	Cultura Estética	20.000			20.000	20.000	
5	Cultura Física	20.000			20.000	20.000	
6	Optativa	20.000			20.000	20.000	
7	Actividades adicionales	13.000			13.000	13.000	
	TOTAL	126.000			126.000	18.000	

Este es un documento en Word que está sujeto a cambios antes de su impresión.

- En caso de que no se quiera imprimir el certificado se hace clic en el botón "Salir" en la ventana que sale al iniciar esta opción.

8. Estadísticas.

Doble clic en "Estadísticas" y se presentaran los ítems:

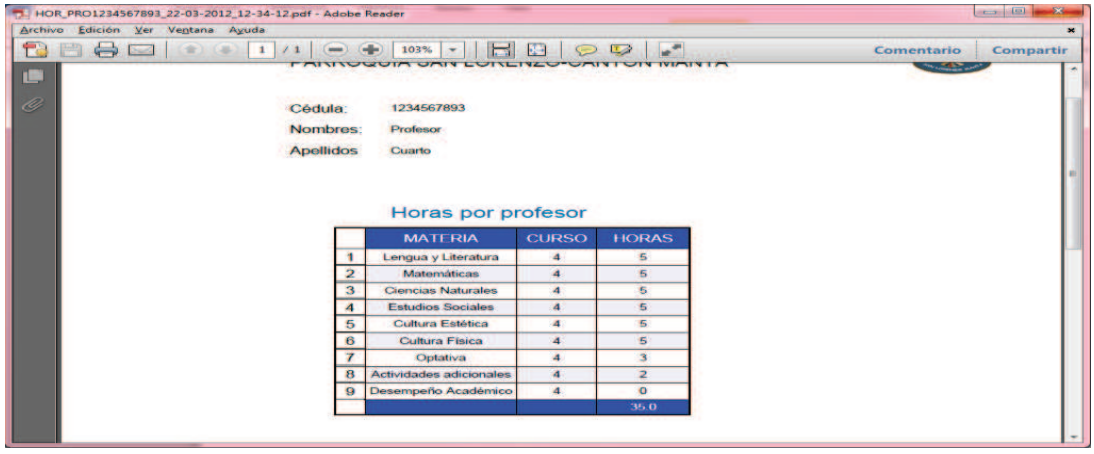
- Horas por profesor.
- Horas por curso.
- Promedios generales.

8.1. Horas por Profesor.

Una vez escogida este ítem en estadísticas se presenta la ventana de "Busqueda General de Profesores" (véase 3. Profesores – 3.1. Búsqueda general) que es esta:



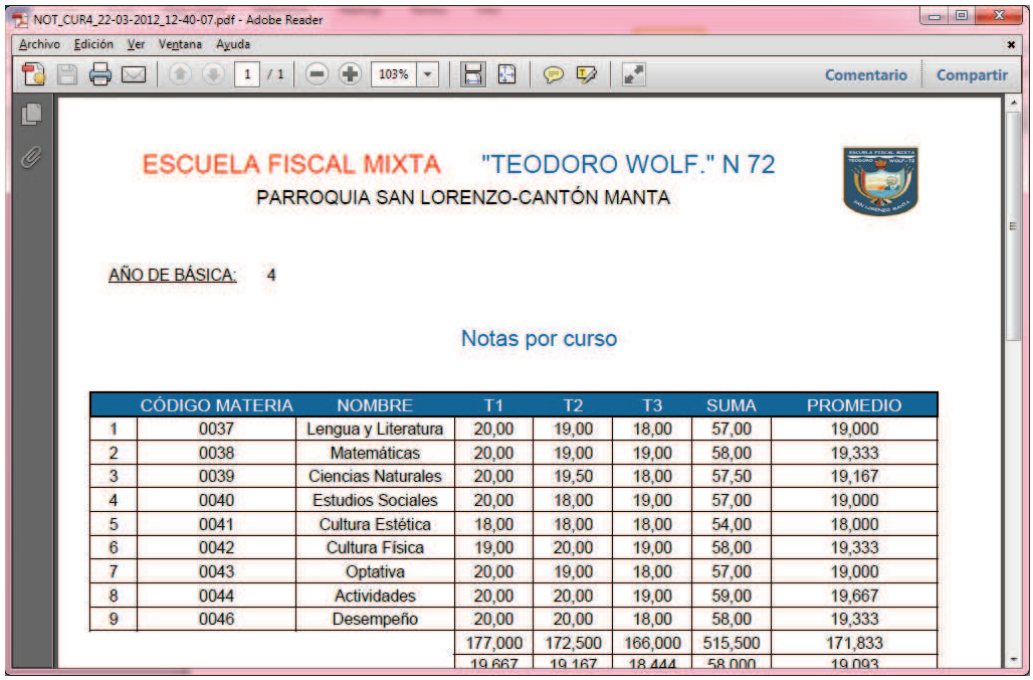
Se escoge el profesor del cual se quieren las estadísticas, se da clic en "Aceptar" en el mensaje emergente que indica la dirección del archivo y se presentara un documento PDF como este:



- En caso de no querer revisar las estadísticas haga clic en el botón "Salir" en la ventana de búsqueda.
- Estas estadísticas pueden ser impresas.

8.2. Notas por Curso.

Esta opción es para ver las notas de un curso en específico. Se escoge el curso en la ventana que surge como en 7. Impresiones – 7.1 Libreta de Calificaciones – 7.1.1. Curso y se da clic en "Aceptar" en el mensaje que aparece indicando la ubicación del archivo. A continuación se muestran las notas, como en este caso:

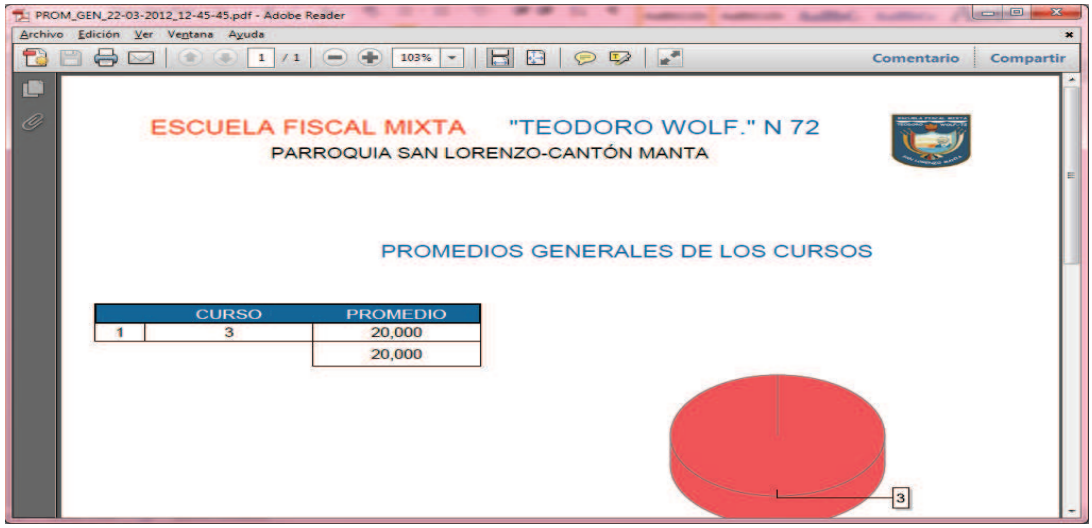


	CÓDIGO MATERIA	NOMBRE	T1	T2	T3	SUMA	PROMEDIO
1	0037	Lengua y Literatura	20,00	19,00	18,00	57,00	19,000
2	0038	Matemáticas	20,00	19,00	19,00	58,00	19,333
3	0039	Ciencias Naturales	20,00	19,50	18,00	57,50	19,167
4	0040	Estudios Sociales	20,00	18,00	19,00	57,00	19,000
5	0041	Cultura Estética	18,00	18,00	18,00	54,00	18,000
6	0042	Cultura Física	19,00	20,00	19,00	58,00	19,333
7	0043	Optativa	20,00	19,00	18,00	57,00	19,000
8	0044	Actividades	20,00	20,00	19,00	59,00	19,667
9	0046	Desempeño	20,00	20,00	18,00	58,00	19,333
			177,000	172,500	166,000	515,500	171,833
			19,667	19,167	18,444	58,000	19,093

- Si no desea ver las notas haga clic en el botón "Salir" que se encuentra en la ventana para escoger el curso.
- Se pueden imprimir las notas en este documento, pero no alterarlas.

8.3. Promedios Generales.

Opción para visualizar los promedios generales de la institución. Al hacer clic en este ítem aparece un mensaje indicando la ubicación del archivo que se mostrará a continuación y que luce algo así:



- Este documento se puede imprimir.
- Este archivo PDF no está sujeto a cambios.

9. Respaldo BBD.

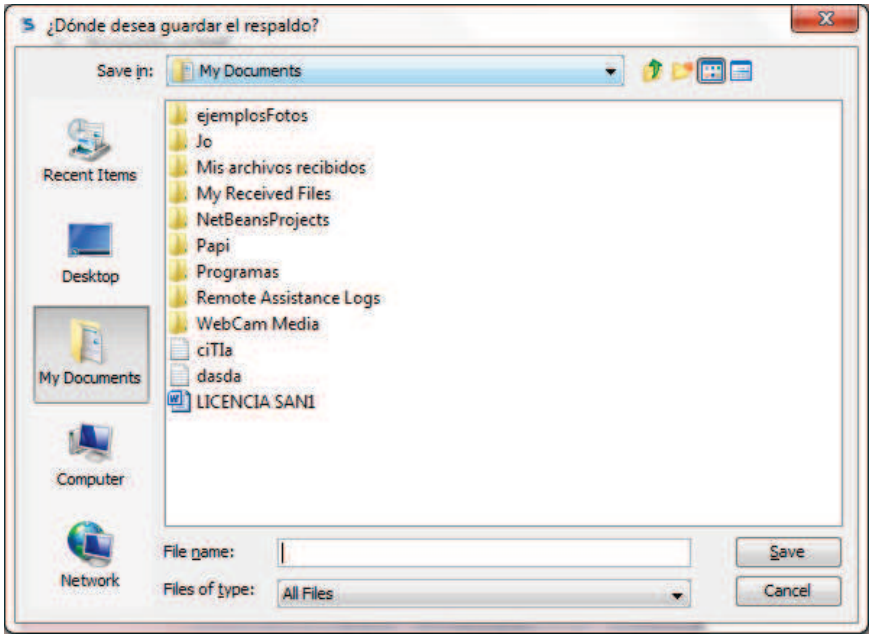
Esta opción fue elaborada en caso de que el sistema sufra alguna falla y su funcionamiento se altere, ya que la información se guarda y así no se perderá.

Al hacer doble clic en "Respaldo BBD" se muestra el ítem:

- Respaldo actual.

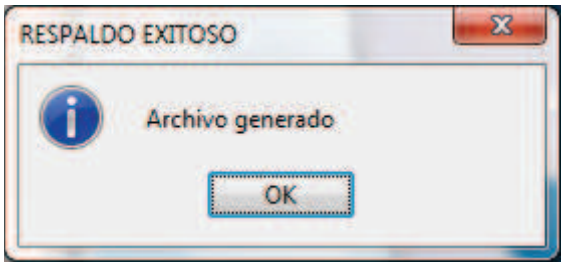
9.1. Respaldo Actual.

Se da clic en "Respaldo actual" y aparece esta ventana:



En la cual usted debe escoger la ubicación donde se guardara el respaldo del programa o ingresar el nombre del archivo nuevo que se creara con motivo de seguridad en el campo denominado "Nombre de archivo:"

- En caso de no querer realizar el respaldo hacer clic en el botón "Cancelar".

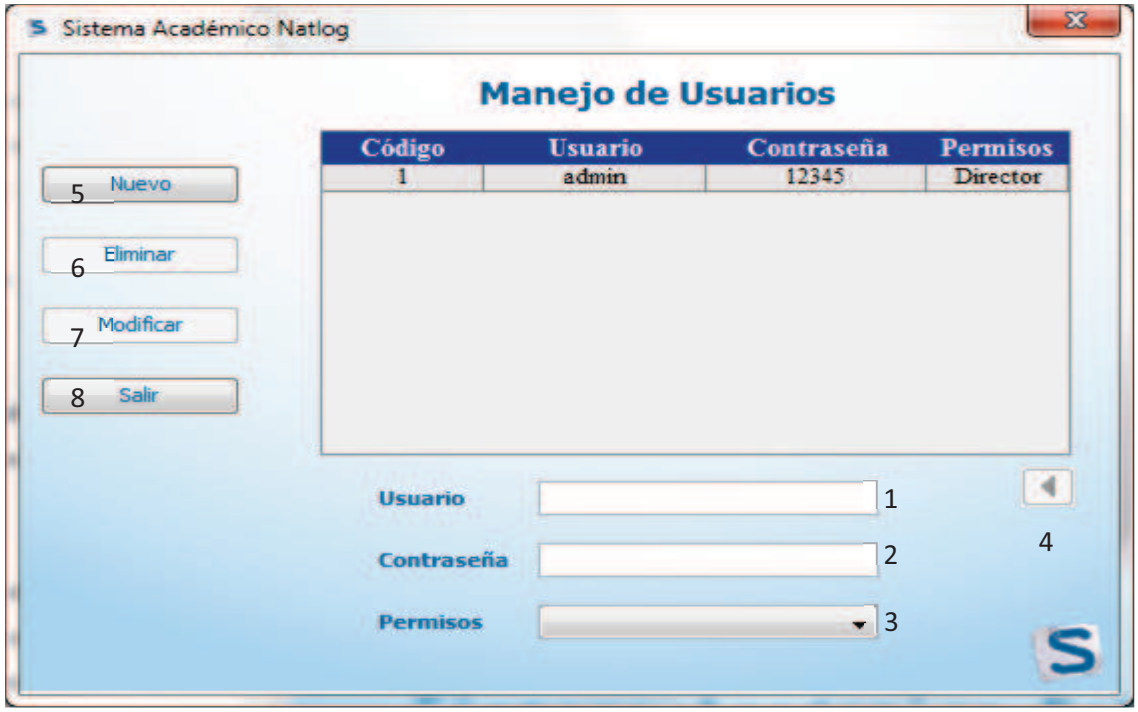
Luego se mostrará este mensaje 

y usted debe dar clic en el botón "Aceptar".

10. Comandos Especiales y Recomendaciones.

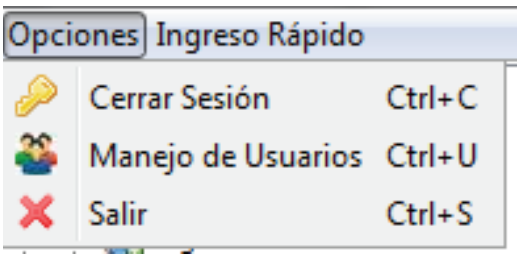
En la ventana principal del programa en la parte superior izquierda hay dos botones: "Opciones" e "Ingreso Rápido".

En su mayoría, estos son atajos a servicios que ya explicamos anteriormente, a excepción de una que se encuentra en "Opciones" y se denomina "Manejo de Usuarios", en la cual al hacer clic aparece esta ventana:



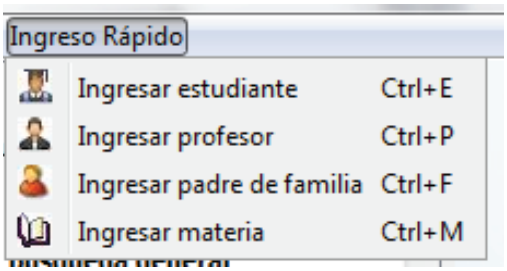
1) Se ingresa el nombre del nuevo usuario.
2) Se ingresa la contraseña.
3) Se escoge entre "Director" que es de uso libre y "Profesor" que es de uso restringido.
4) Refresca la pantalla ya que cuando se ha seleccionado un usuario se desactiva la posibilidad de crear uno nuevo y con este botón se vuelve a activar.
5) Botón al que se da clic una vez ingresados los datos del nuevo usuario y automáticamente se crea.
6) Botón que se presiona después de escoger un usuario al que se desea eliminar.
7) Se hace clic en esa opción después de escoger un usuario al que se le desean realizar cambios.
8) Se hace clic aquí cuando se desea finalizar el manejo de usuarios.

Los atajos que se presentan en "Opciones" e "Ingreso Rápido" se agilizan con los siguientes comandos:



Opciones.

- ✓ **Cerrar Sesión; Ctrl + C**
- ✓ **Manejo de Usuarios: Ctrl + U**
- ✓ **Salir: Ctrl + S**



Ingreso Rápido.

- ✓ **Ingresar estudiante: Ctrl + E**
- ✓ **Ingresar profesor: Ctrl + P**
- ✓ **Ingresar padre de familia: Ctrl + F**
- ✓ **Ingresar materia: Ctrl + M**

Recomendaciones.

- ❖ Llenar todos los campos aunque no sean obligatorios, así el sistema tendrá información más completa.
- ❖ Seguir el manual ya que su fin es facilitar el uso de nuestro programa y evitar inconvenientes con él.
- ❖ Ingresar en el sistema solo información verídica ya que el proveer información errónea lo único que lograra es perjudicar a la institución y su credibilidad.
- ❖ Recordar que las fotos solo pueden tener un máximo de tamaño de 300kb.

ANEXO C

SISTEMA ACADEMICO NATLOG "S.A.N."

Manual técnico.

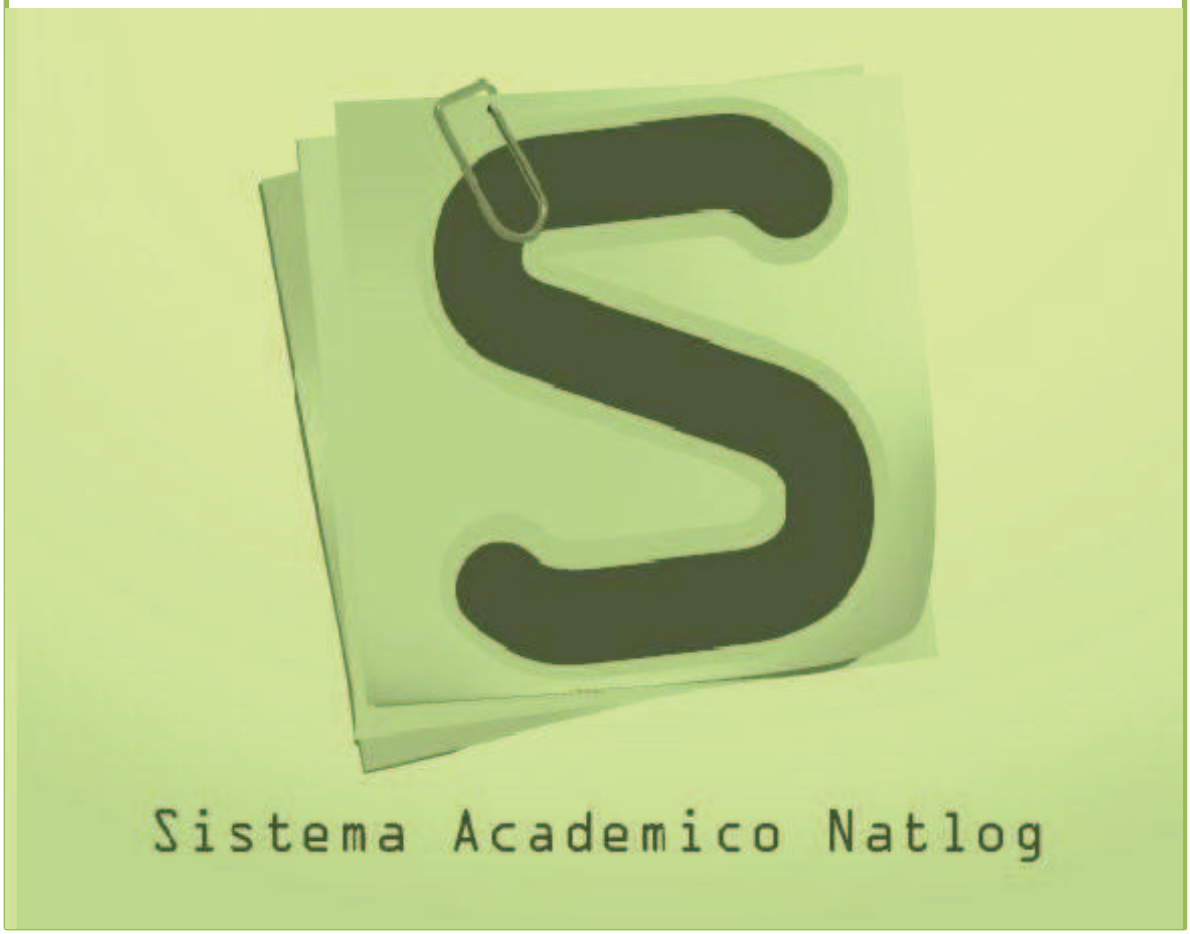


Tabla de Contenidos.

11. Indicaciones iniciales..……..……1
12. Instalación de Java……2
13. Instalación de MySQL …...……..……..…..7
14. Instalación de Sistema Académico Natlog…..…….…….……..15
15. Creación de la base de datos……...…18
16. Plantillas de reportes……...……24
17. Cargar un respaldo de base de datos……...……26
18. Indicaciones finales…..……...33

3. Indicaciones iniciales.

El Sistema Académico Natlog (SAN) esta escrito en el lenguaje JAVA y utiliza como motor de base de datos MySQL 5.5 es por esto que son requerimientos necesarios para poder funcionar de una buena manera.

Además se requiere ejecutar un script de creación de la base de datos que contiene todas las tablas que se necesitan, incluyendo también los datos iniciales que se van a utilizar.

Por ultimo para generar reportes se debe copiar las plantillas a su computador. Todo esto será explicando punto a punto.

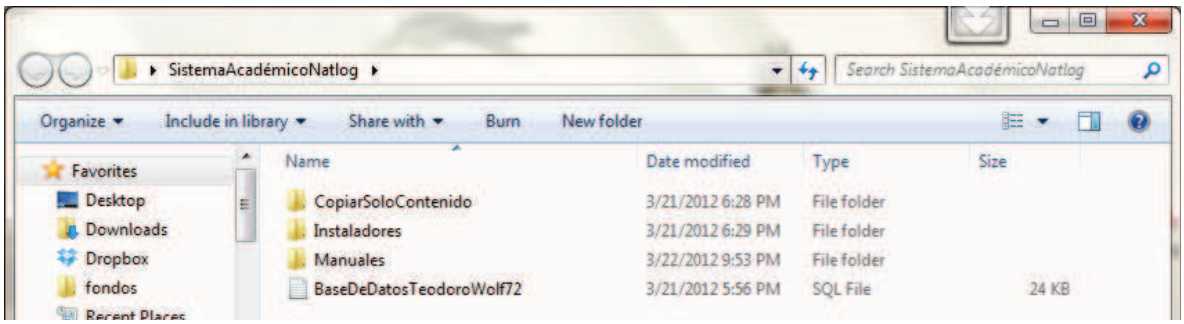
Comencemos:

Una vez con el CD de Instalación de SAN, lo colocamos en el CD-ROM. Esperamos a que cargue y procedemos a explorarlo



La carpeta que contiene el CD es la siguiente

SistemaAcadémicoNatlog

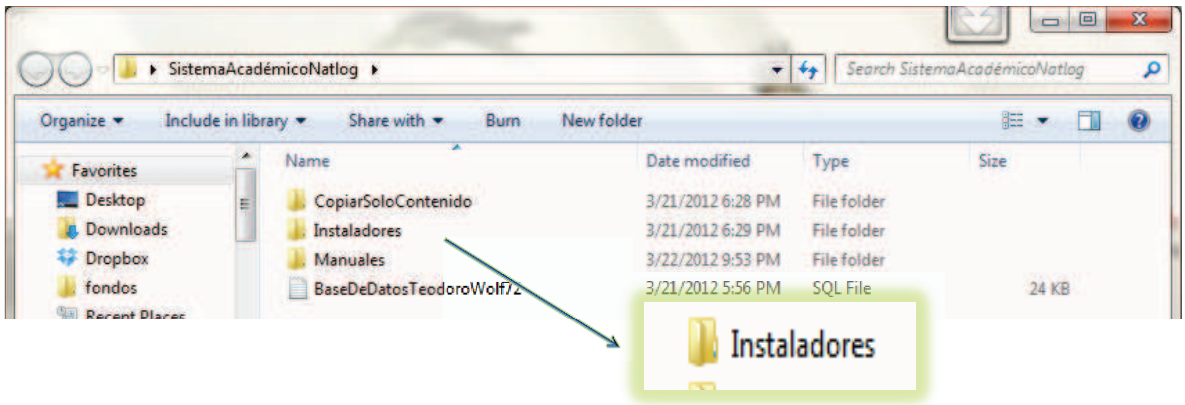


Con esta carpeta es con la que se va trabajar para la instalación.

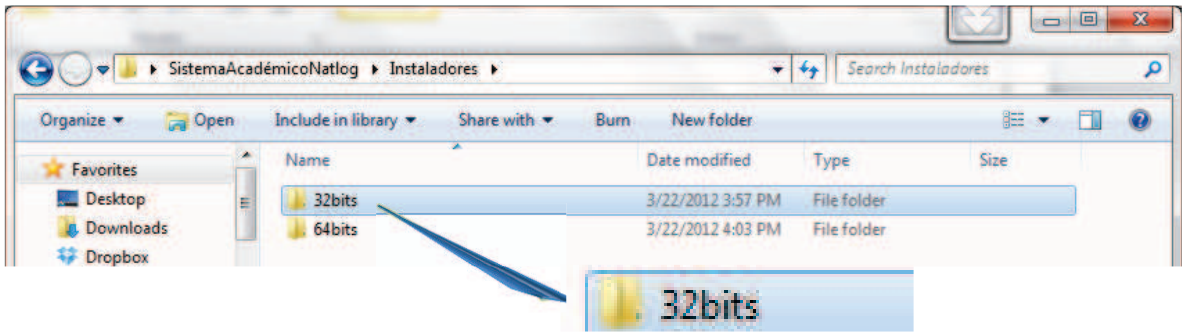
2. Instalación de Java.

Java es el lenguaje de programación en el que fue desarrollado el aplicativo SAN. Por esta razón es fundamental su instalación. Este manual de java ha sido tomado en parte de la página http://www.luiskano.net/. Ya que nos compete es la instalación de SAN.

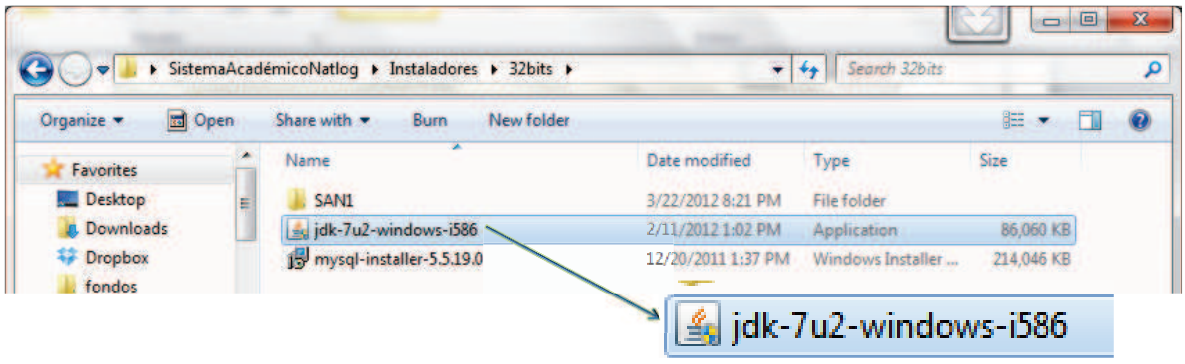
1. Abrimos la carpeta Instaladores



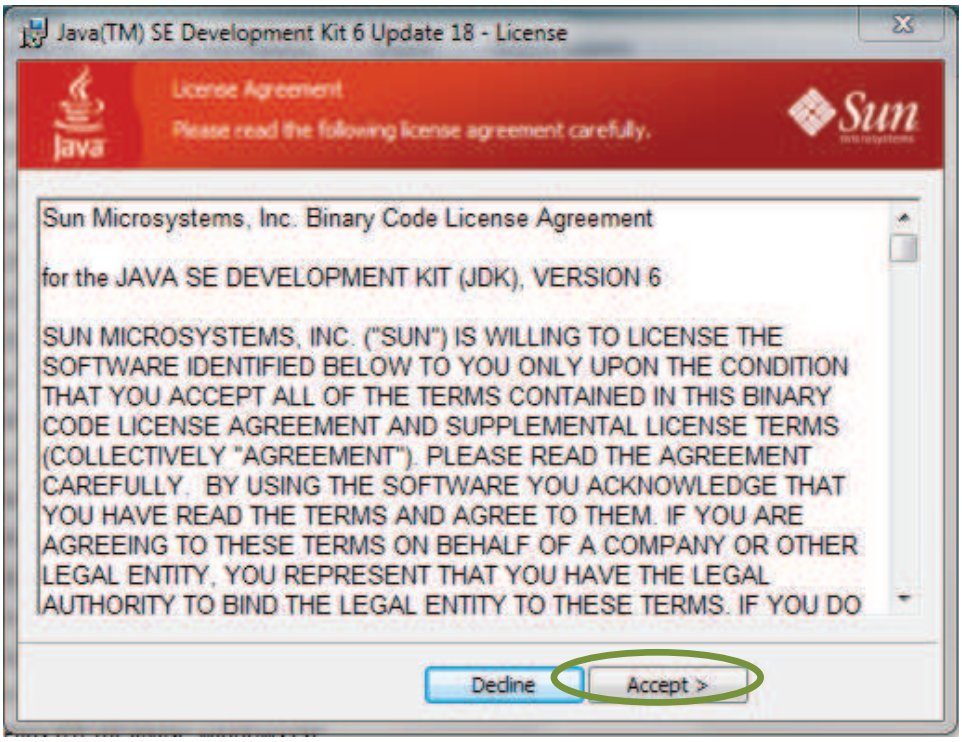
2. Dependiendo de nuestro Sistema Operativo escogemos la opción de 32 bits o de 64 bits, para efecto de este manual vamos a escoger 32bits.



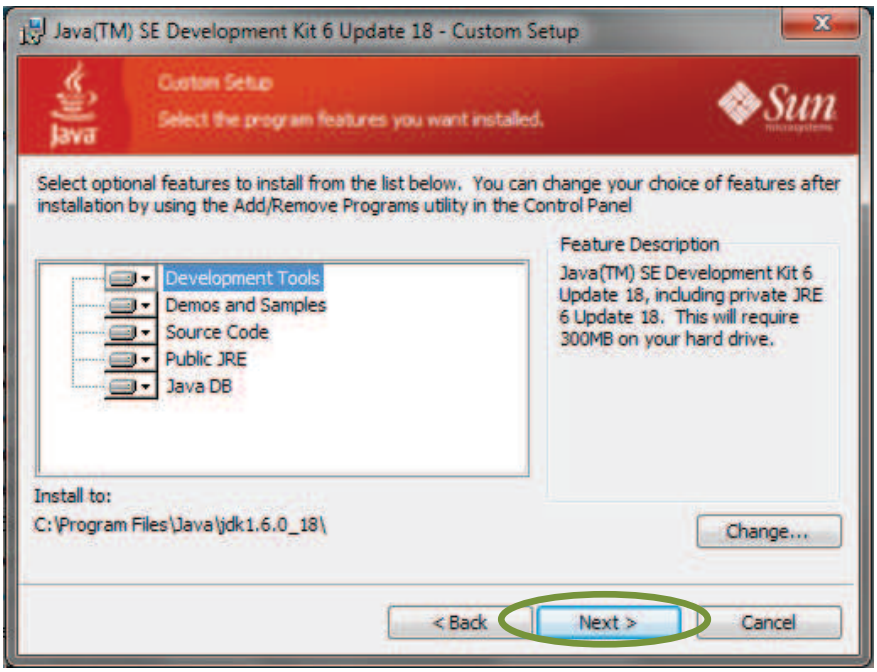
3. Para comenzar la instalación escogemos el archivo: jdk-7u2-windows-i586



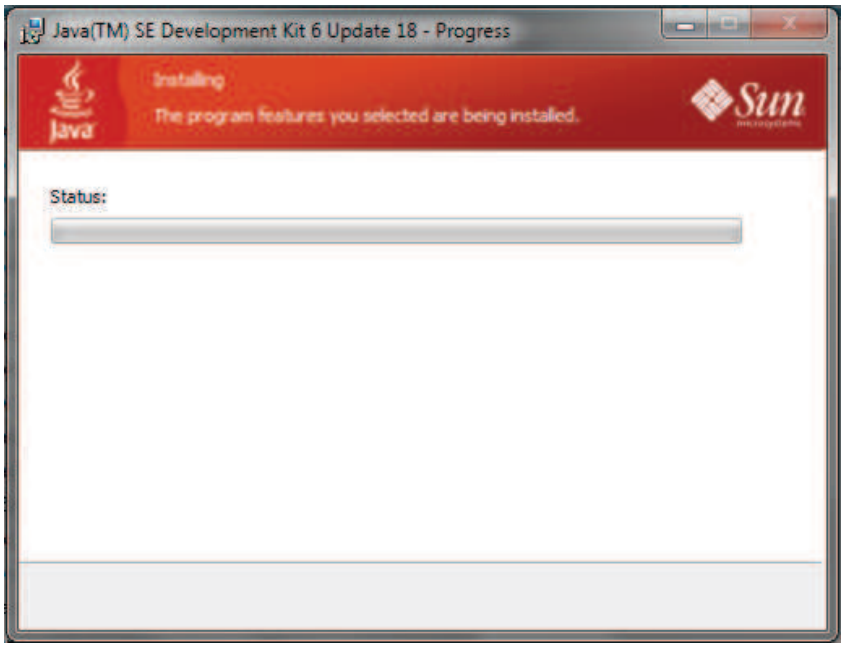
4. Y nos aparecerá una pantalla donde leemos **el contrato de** Usuario **Final,** y le damos Clic en **Accept >**



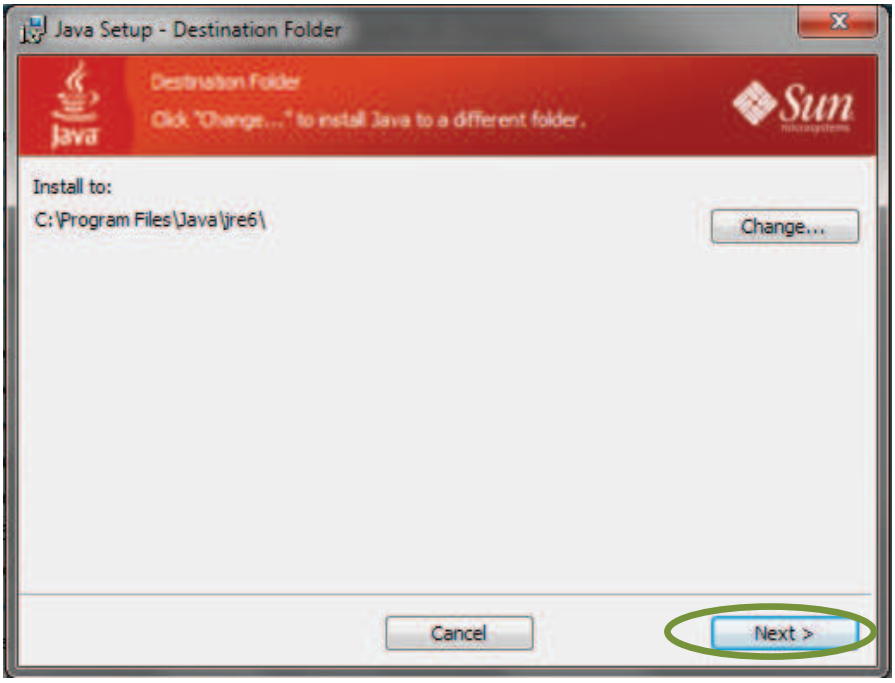
5. Si queremos cambiar la capeta de instalación le podemos dar clic en **Change…** e instalarla donde se quiera, se recomienda dejar en la que está por default. Después le damos clic en **Next >**.



6. **Empezará la instalación del Java JDK** y tendremos que esperar unos minutos, dependiendo del rendimiento de tu equipo de cómputo**.**



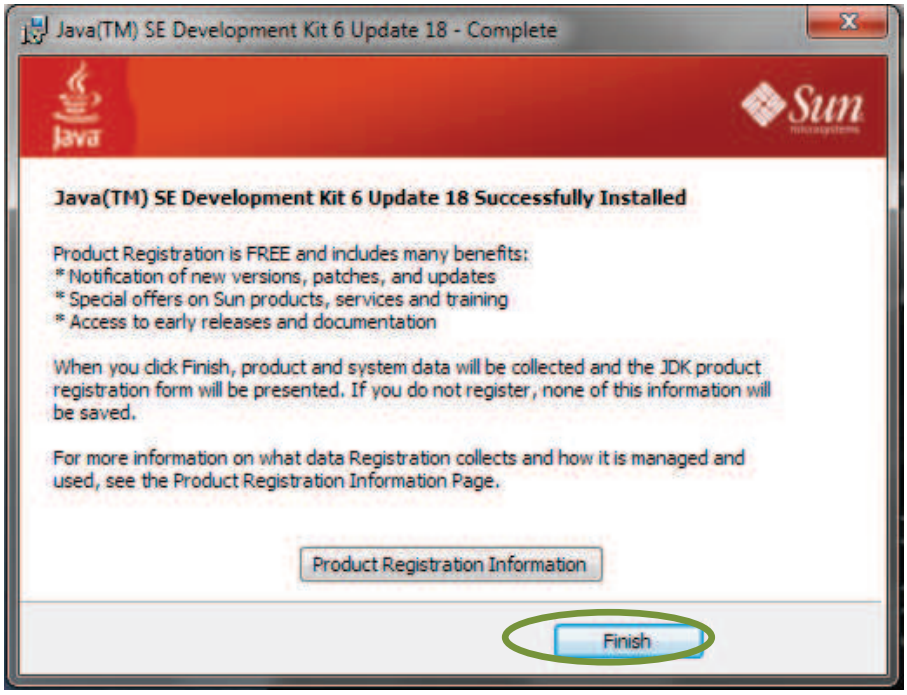
7. Al terminar la instalación del **Java JDK**, nos pedirá la carpeta de destino donde se tiene que instalar el **JRE de Java,** este es importante instalarlo, para que se ejecuten los programas hecho en Java. Se recomienda dejar la ruta en la que está por default. **Damos Clic en Next >**.



8. Empezará la instalación de Java JRE y dependerá del funcionamiento de nuestro equipo, cuanto dure esta instalación, por lo general dura unos pocos minutos.

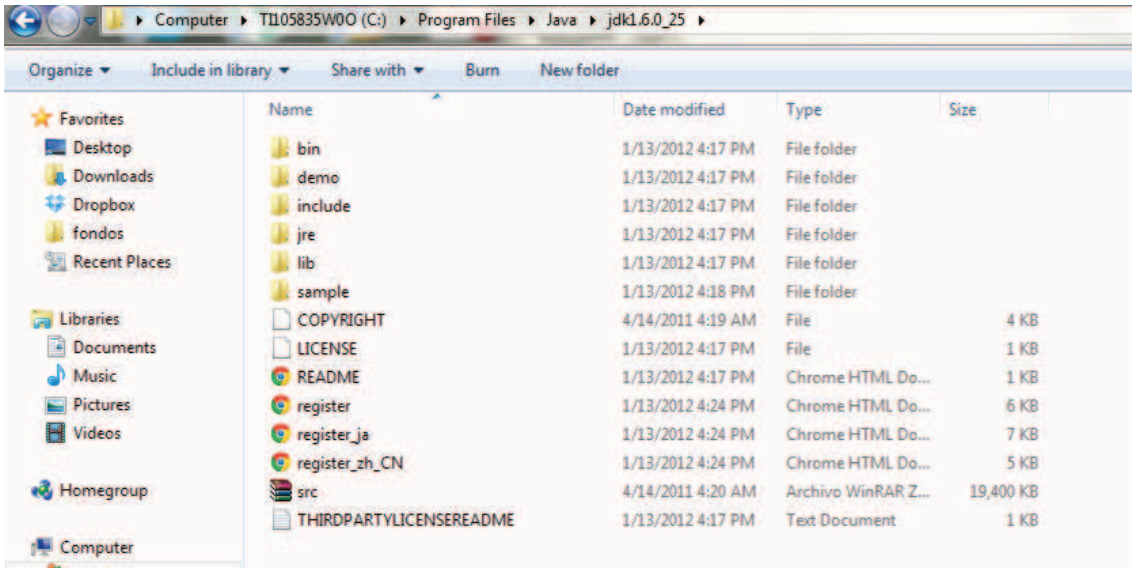


9. Una vez que haya terminado la instalación del **Java JRE**, le daremos clic en **Finish** y terminará la instalación de **Java JDK y Java JRE.**



10. Comprobamos que se haya instalado correctamente en

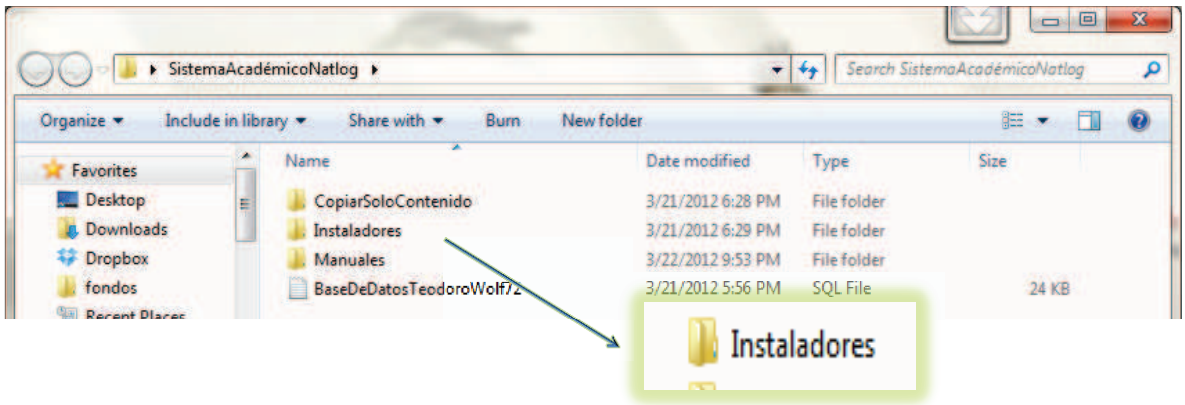
C:\Archivos de programas\Java\jdk1.6.0_25 (Program Files si esta en inglés)



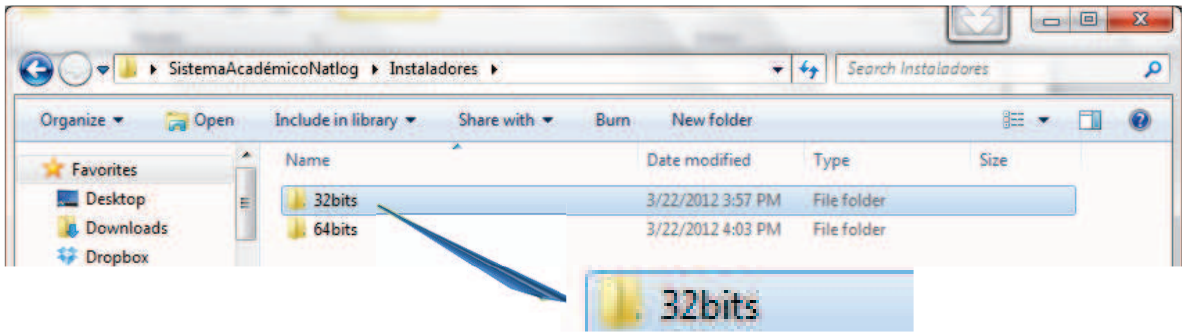
3. Instalación de MySQL 5.5

MySQL es el motor de base de datos que utiliza SAN, por esto se debe instalar para poder manipular toda la información que se desee ingresar. A continuación los pasos a seguir para su instalación:

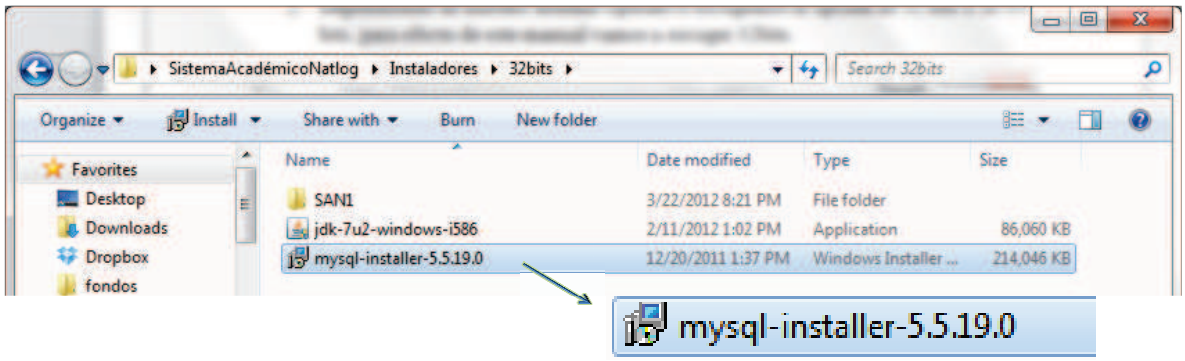
1. Abrimos la carpeta Instaladores



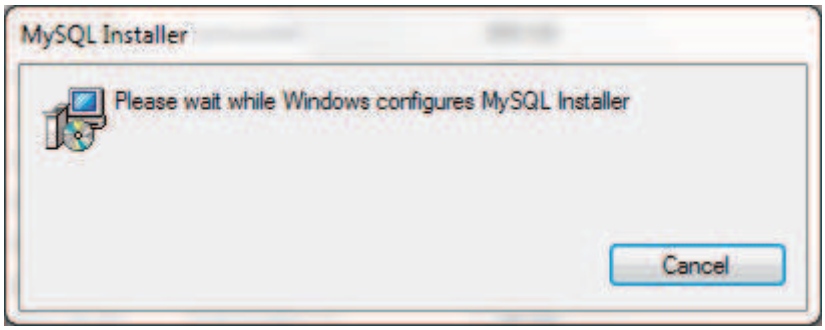
2. Dependiendo de nuestro Sistema Operativo escogemos la opción de 32 bits o de 64 bits, para efecto de este manual vamos a escoger 32bits.



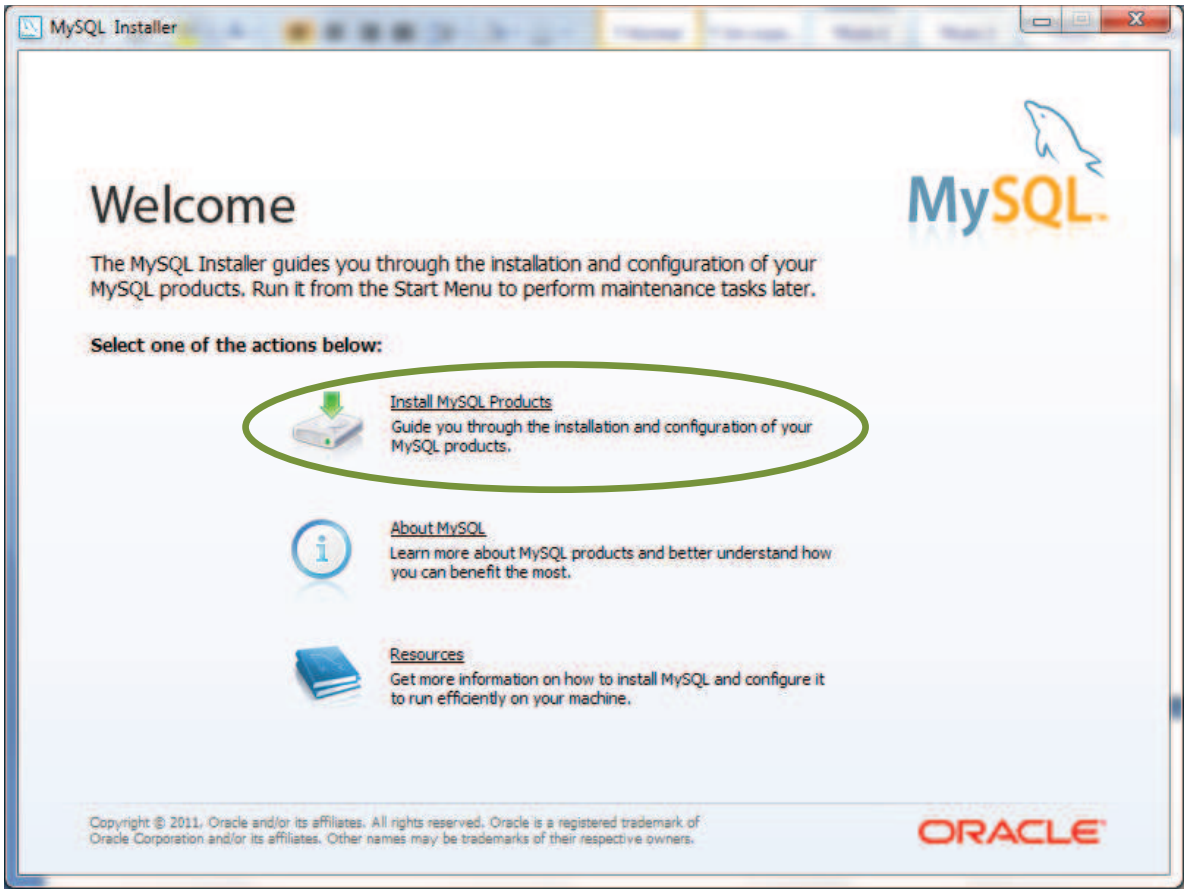
3. Para comenzar la instalación escogemos el archivo: mysql-installer-5.5.19.0



4. Una que le hemos dado doble clic al archivo comienza la instalación. Dependiendo



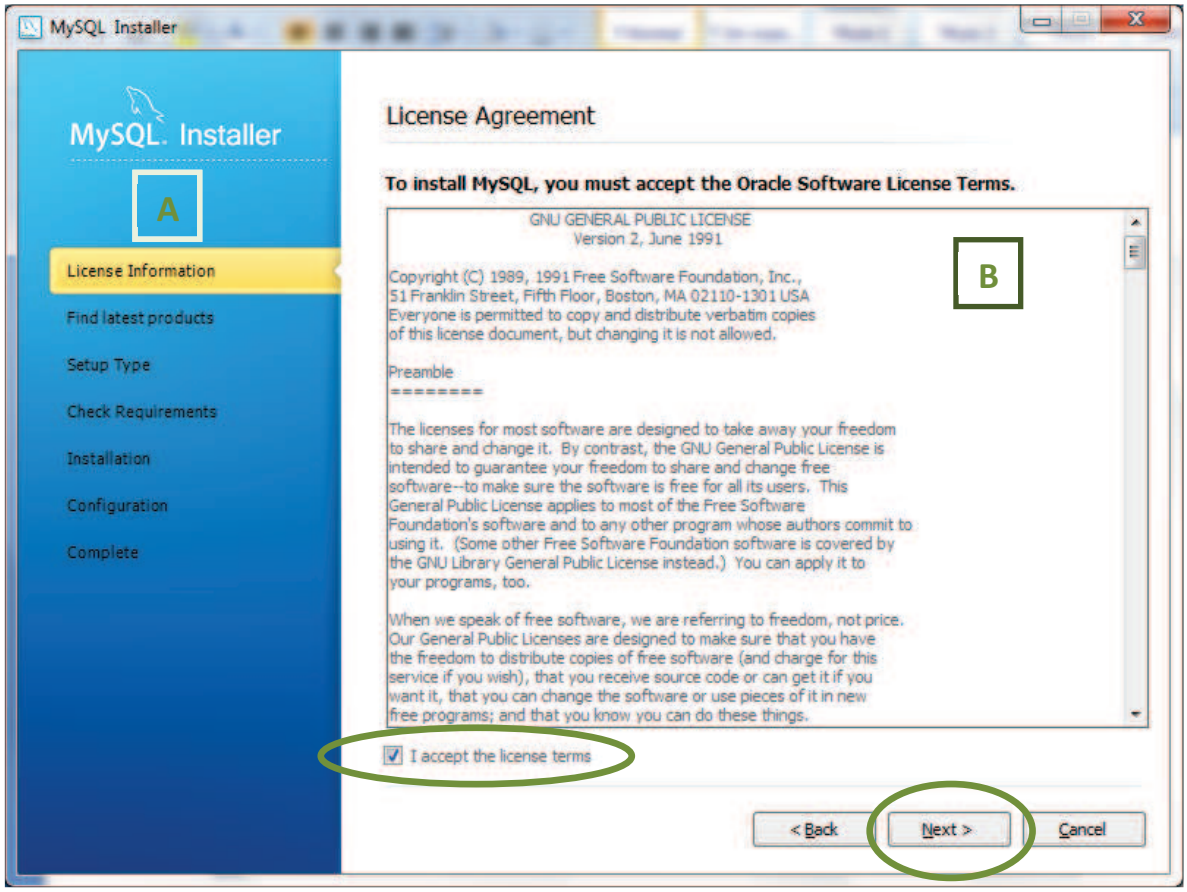
5. Escogemos la opción **Install MySQL Products**



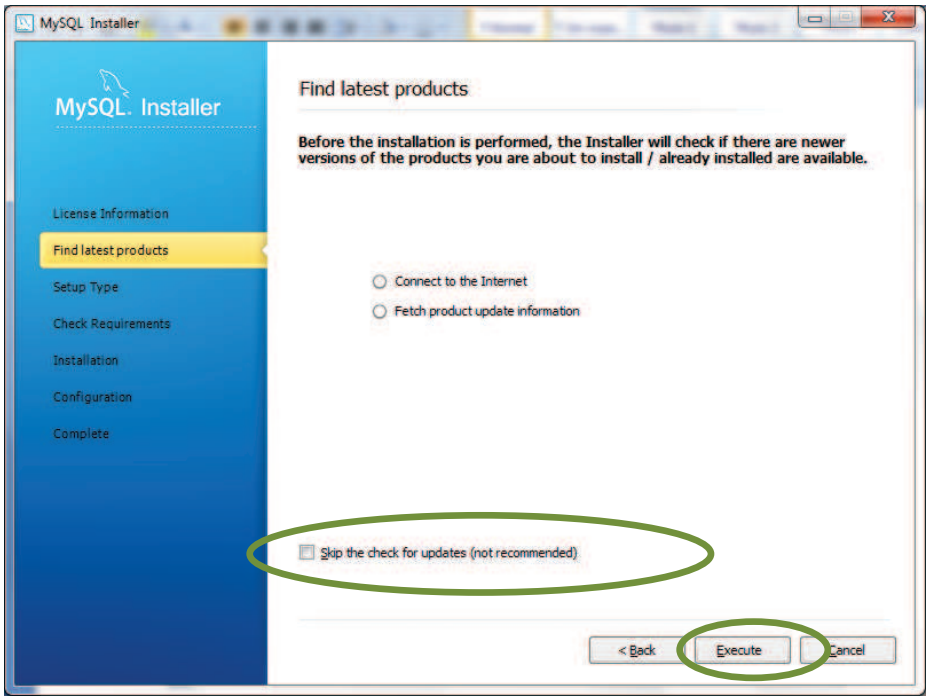
6. Esta es la pantalla principal de instalación con las siguientes opciones:

A: Listado de todos los pasos a seguir durante la instalación
B: Licencia del producto

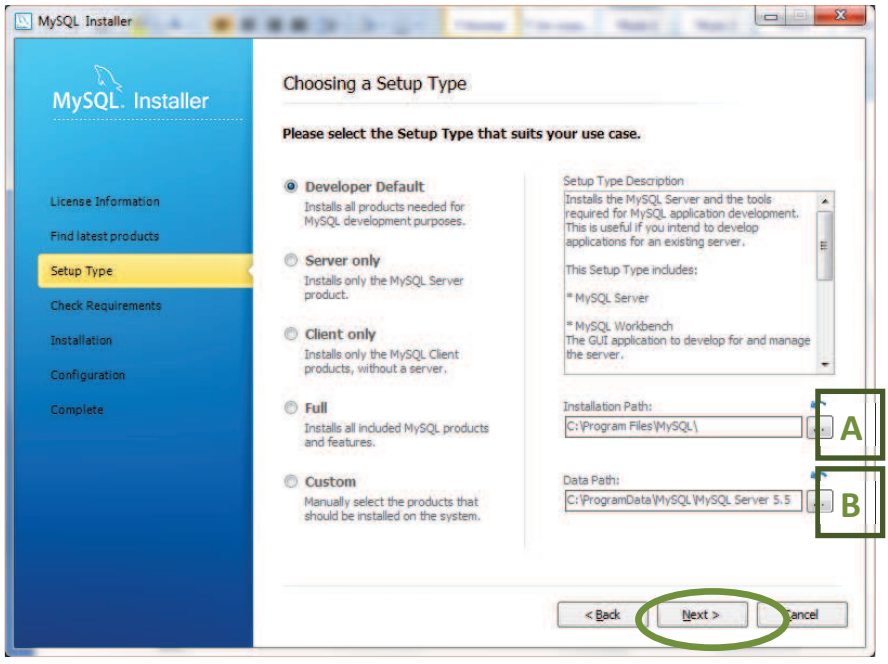
Seleccionamos la opción de ***I accept the license terms*** y dar click en botón **Next>**



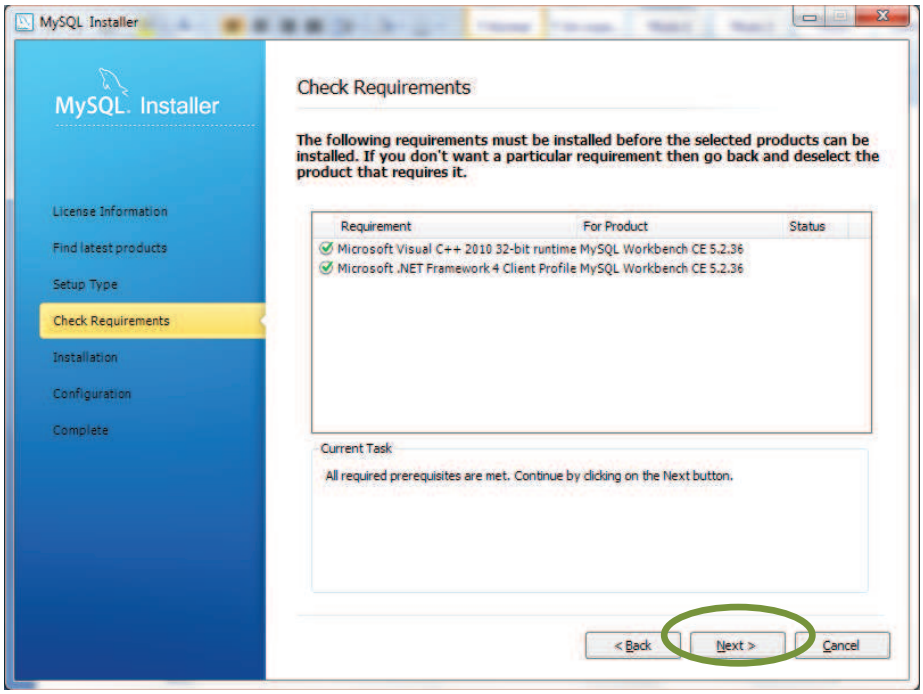
7. La opción ***Skip the check updates (not recommended)*** debe estar marcada (con el visto) y luego damos clic en el botón **Execute**



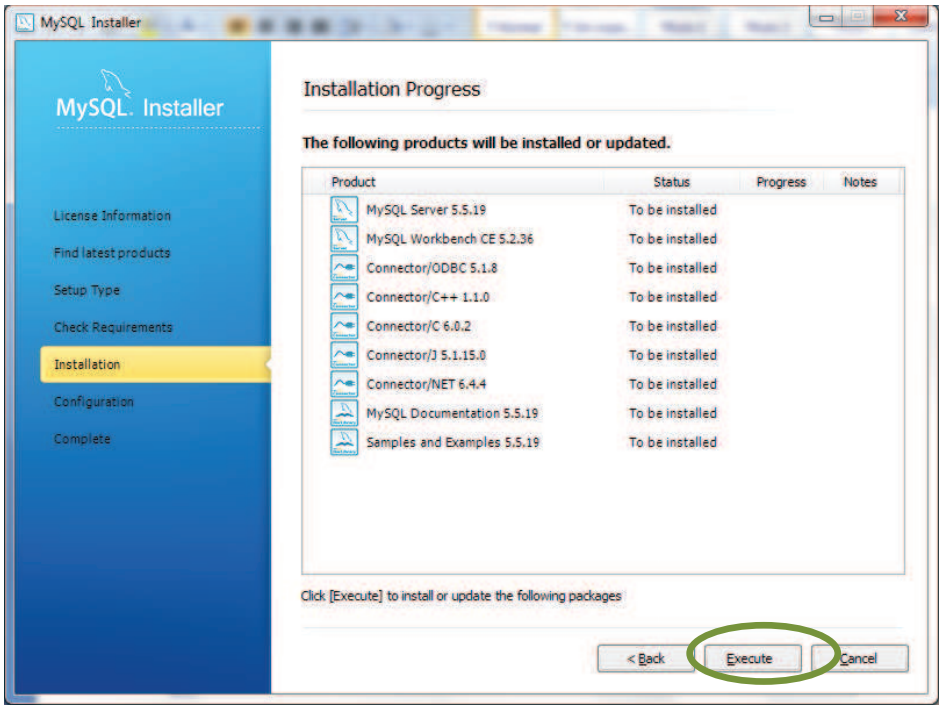
8. En esta ventana dejamos las opciones tal y como están y seleccionamos Next>.
   A: Para cambiar la carpeta de instalación del programa (No cambiar)
   B: Para cambiar el lugar donde se almacenan los datos (No cambiar)



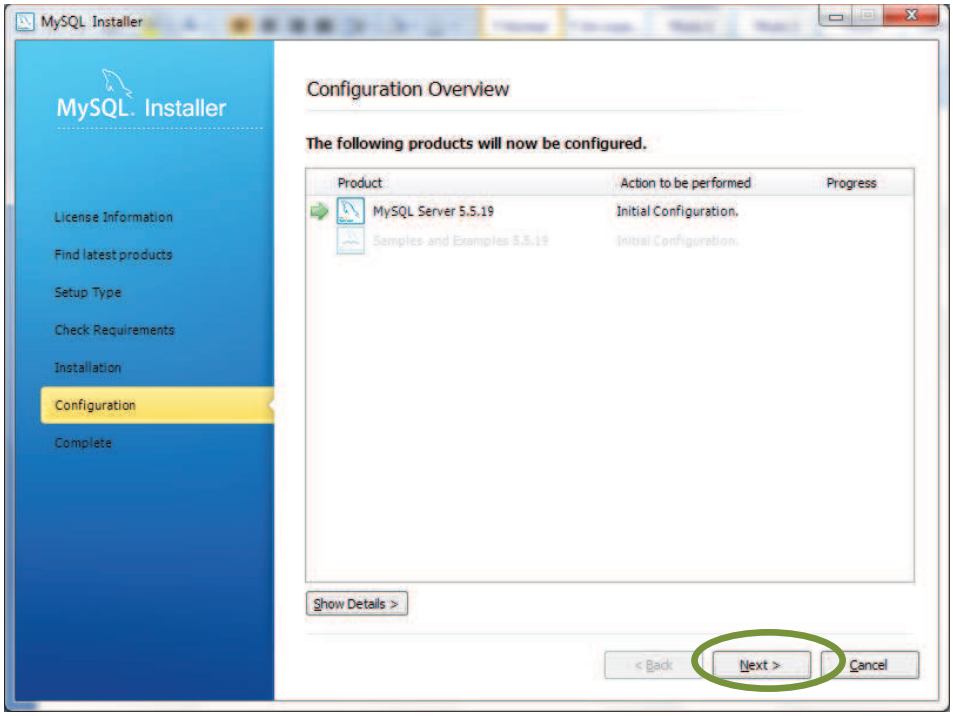
9. MySQL 5.5 requiere de Visual C++2010 y el Framework 4 los dos de Microsoft, en caso de no tenerlo se comienzan a descargar solos. Se requiere de una conexión a internet para realizarlo. Luego de esto se da clic en **Next>**



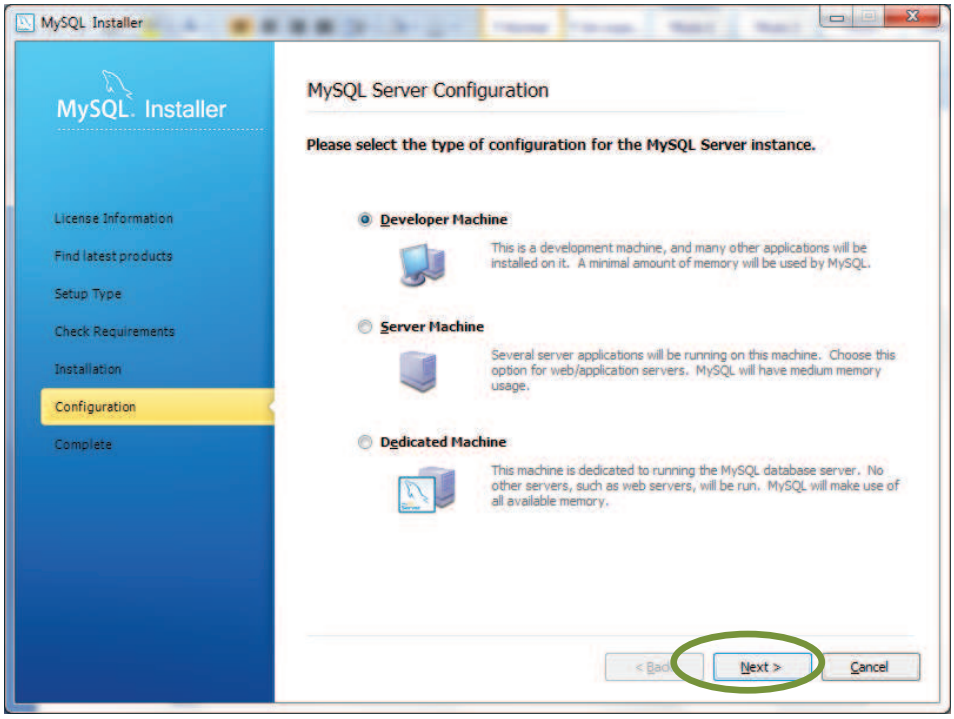
Podemos observar todos los productos que se van a instalar. Dar clic en **Execute**



10. Luego de haberse instalado todos los productos nos aparece la siguiente ventana donde vamos a configurar la base de datos. Damos clic en **Next>**



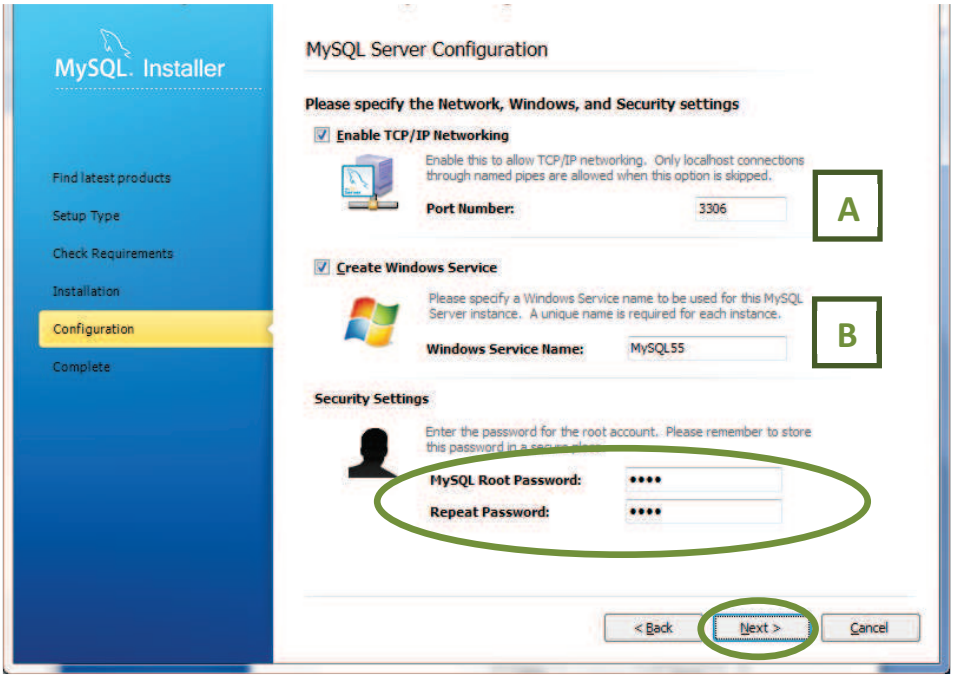
11. Dejamos en la opción ***Developer Machine*** y presionamos **Next >**



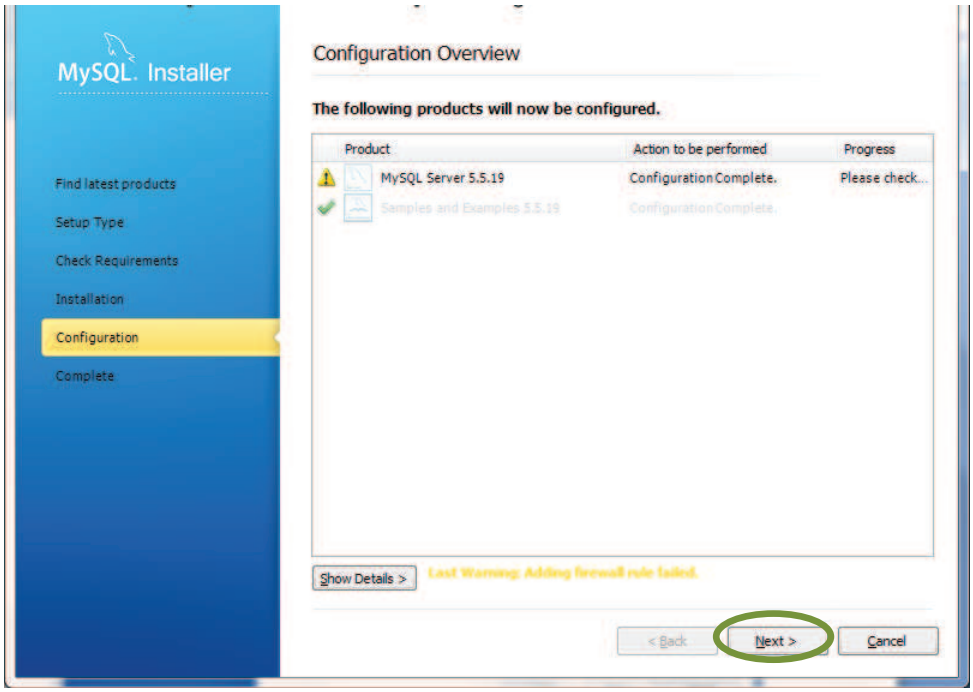
12. En esta ventana configuraremos nuesta base de datos, lo unico que debos escribir es la clave de usuario y luego repetirla. Los otros campos no debemos cambiarlos. Para el correcto funcionamiento del aplicativo SAN se debe escribir la contrasena "root"(sin las comillas) en ambas casillas y damos clic en **Next>**

A: Para cambiar el puerto donde se levantara el servicio (No cambiar)

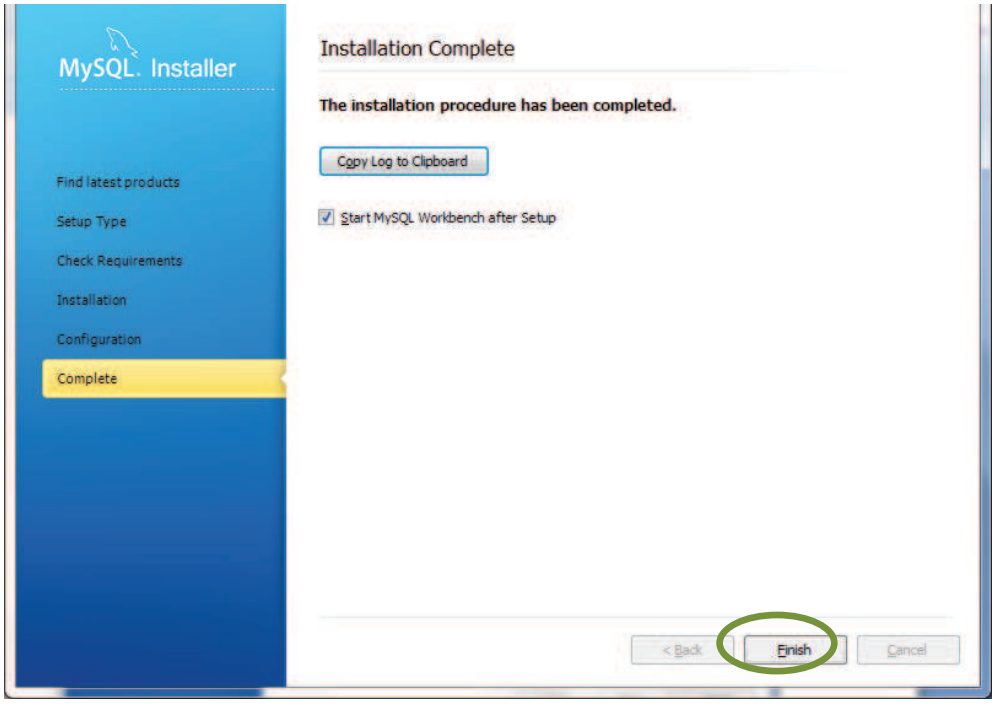
B: Para cambiar el nombre del Servicio (No cambiar)



13. Luego nos aparece esta ventana donde damos clic en **Next>.** Se cargalos ejemplos de SQL y volvemos a dar clic en **Next>**



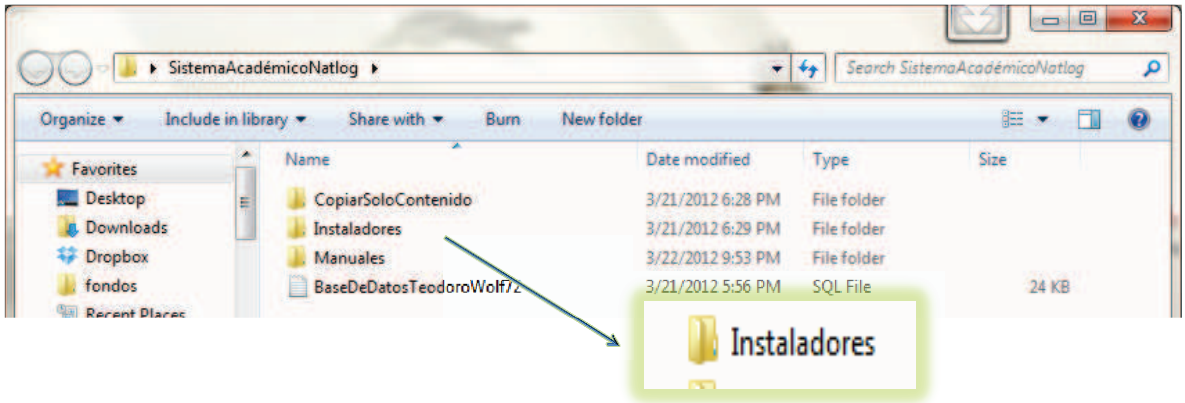
14. Con eso culmina la instalación y damos clic en **Finish**



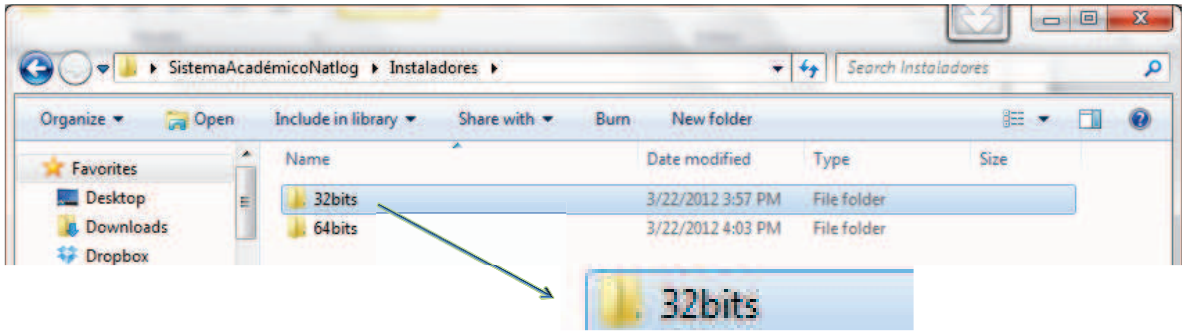
4. Instalación de Sistema Académico Natlog.

Luego de haber instalado Java y MySQL procedemos a la instalación del Sistema Académico Natlog. Se deben realizar los siguientes pasos:

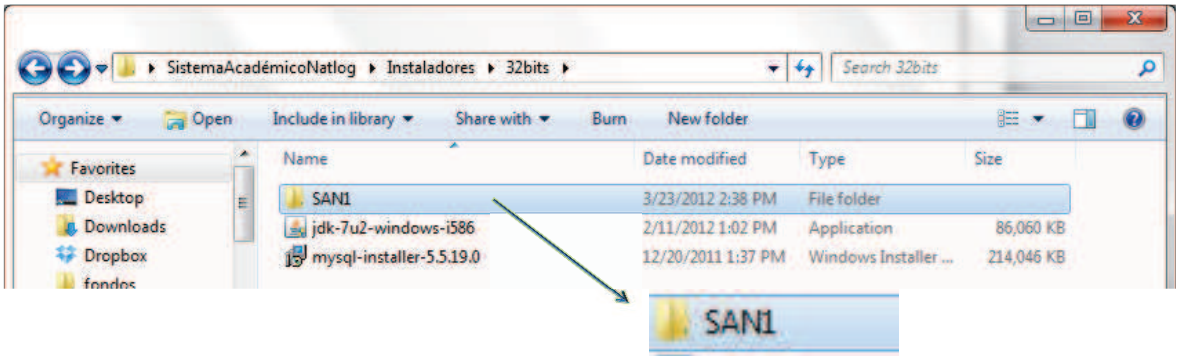
1. Abrimos la carpeta Instaladores



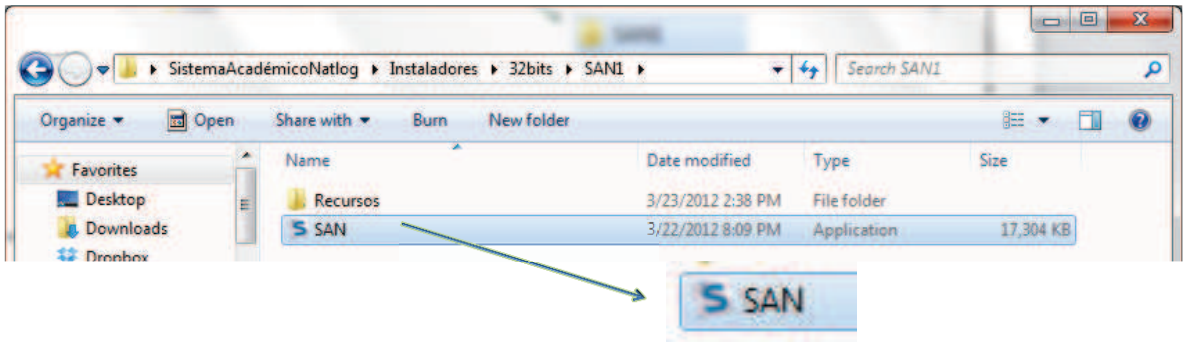
2. Dependiendo de nuestro Sistema Operativo escogemos la opción de 32 bits o de 64 bits, para efecto de este manual vamos a escoger 32bits.



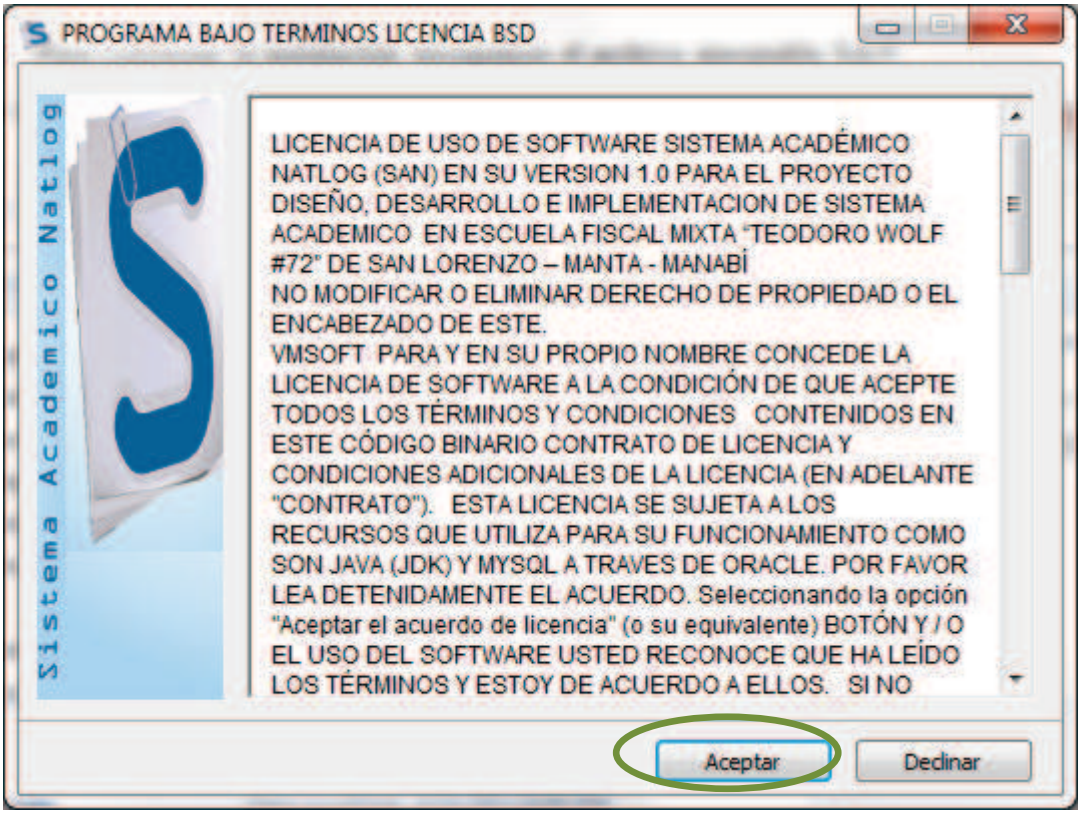
3. Escogemos la carpeta



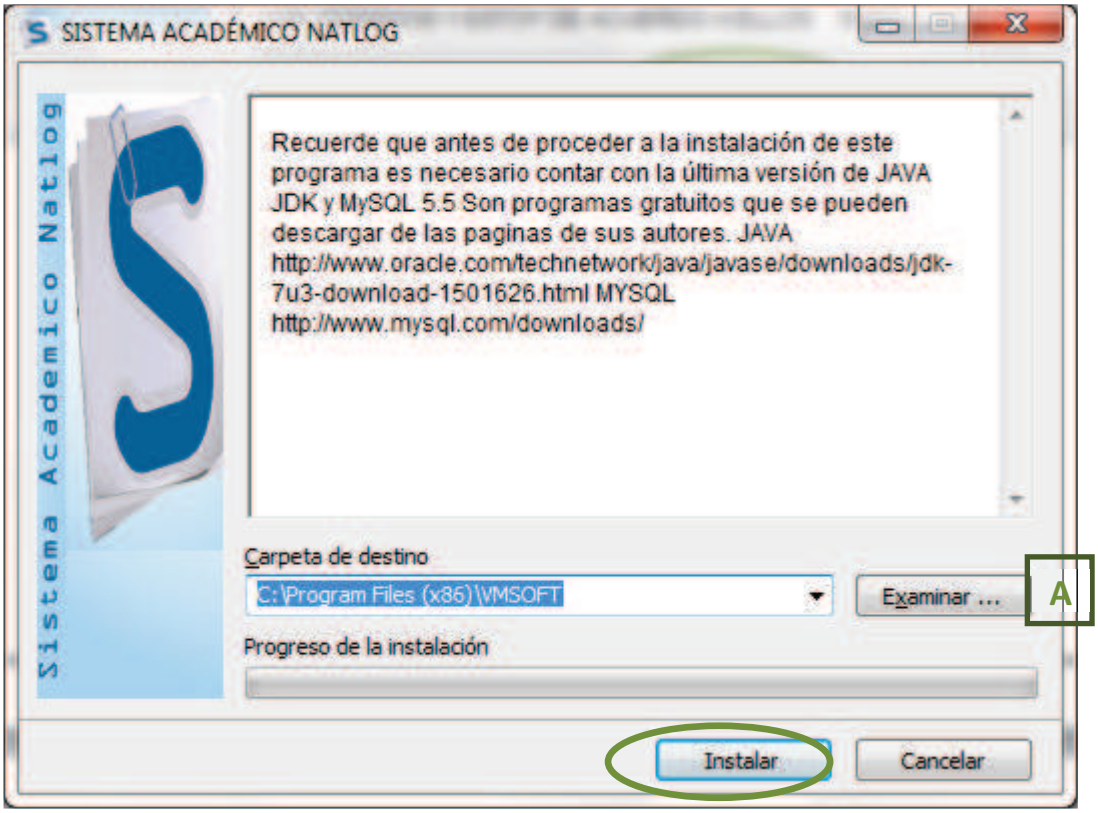
4. Para comenzar la instalación escogemos el archivo ejecutable SAN



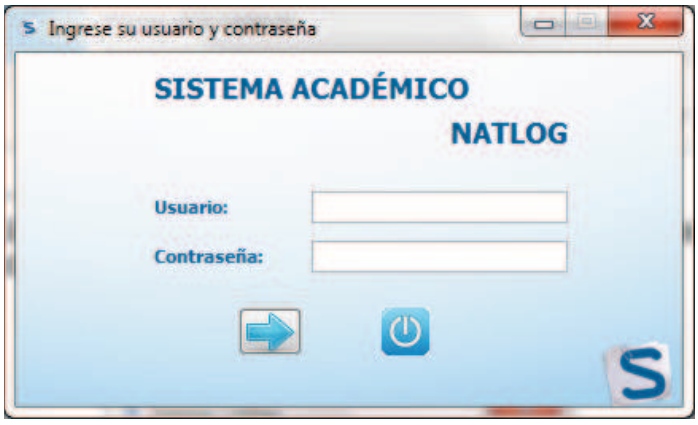
5. La ventana que nos aparecerá es la siguiente. Recordar que SAN se encuentra bajo los términos de la licencia BSD. Para aceptar este acuerdo presiona en Aceptar



6. Para poder utilizar este programa se debió haber instalado Java y MySQL, como también lo contempla este manual. Damos clic en **Instalar**
   A: Si queremos cambiar la ubicación del lugar donde se va a instalar el programa. Por default la vamos a dejar ahí.



7. Luego se instala la aplicación y nos da la bienvenida al programa. Se crea un acceso directo en el Escritorio
   NOTA: Recordar que se debe seguir hasta el final el manual para que funcione correctamente el aplicativo

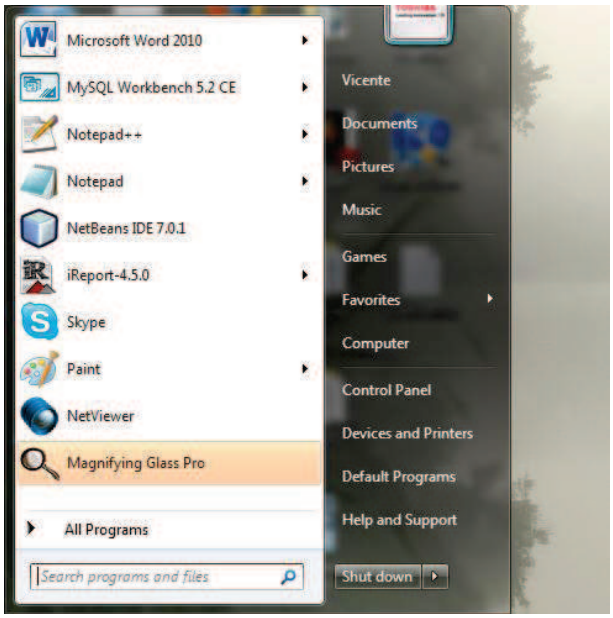




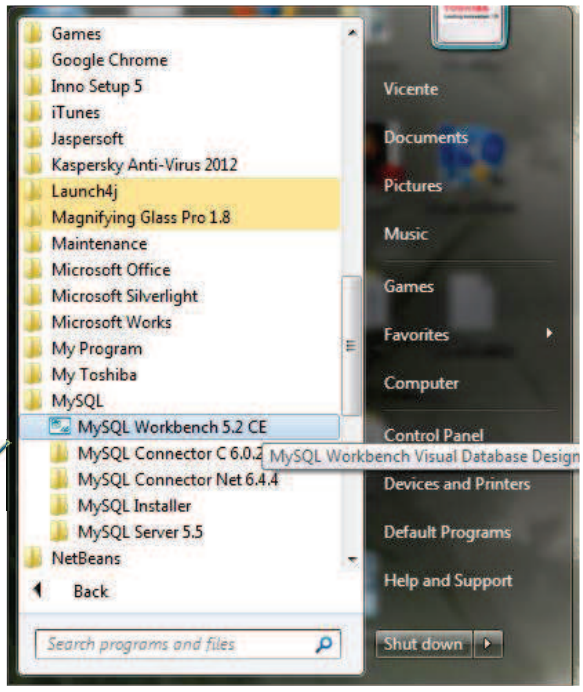
5. Creación de la base de datos.

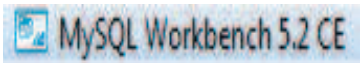
SAN necesita una base de datos para funcionar, es por esto que se incluye un archivo mediante el cual se va a realizar la creación de las tablas donde se va a insertar, borrar y eliminar la información que se vaya proporcionando. Para correr este archivo de creación de base de datos se deben seguir los siguientes pasos.

1. Nos localizamos en Inicio. Luego escogemos "Todos los programas" (All Programs cuando el sistema operativo esta en inglés).

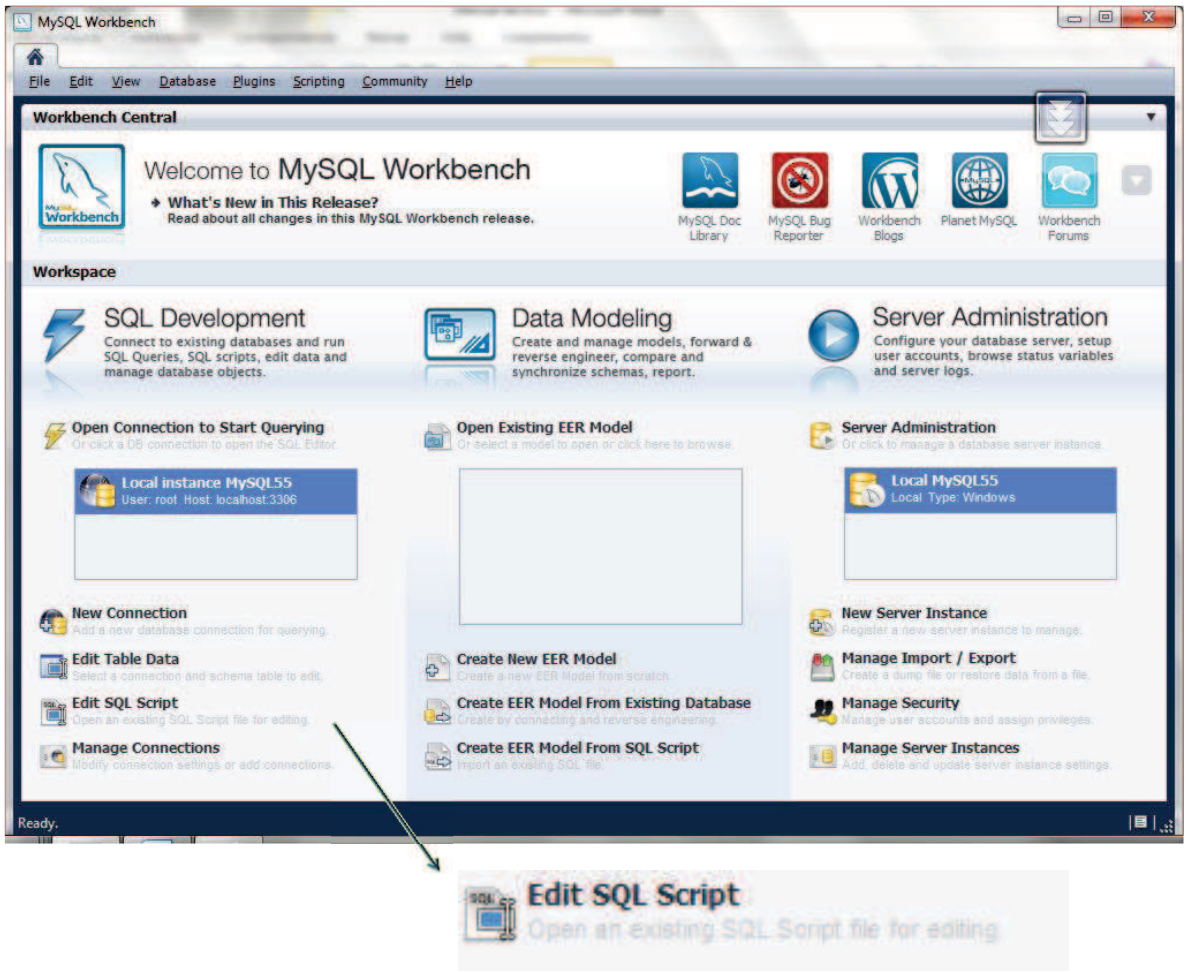


2. Buscamos la carpeta MySQL y damos clic en MySQL Workbrenach 5.2 CE

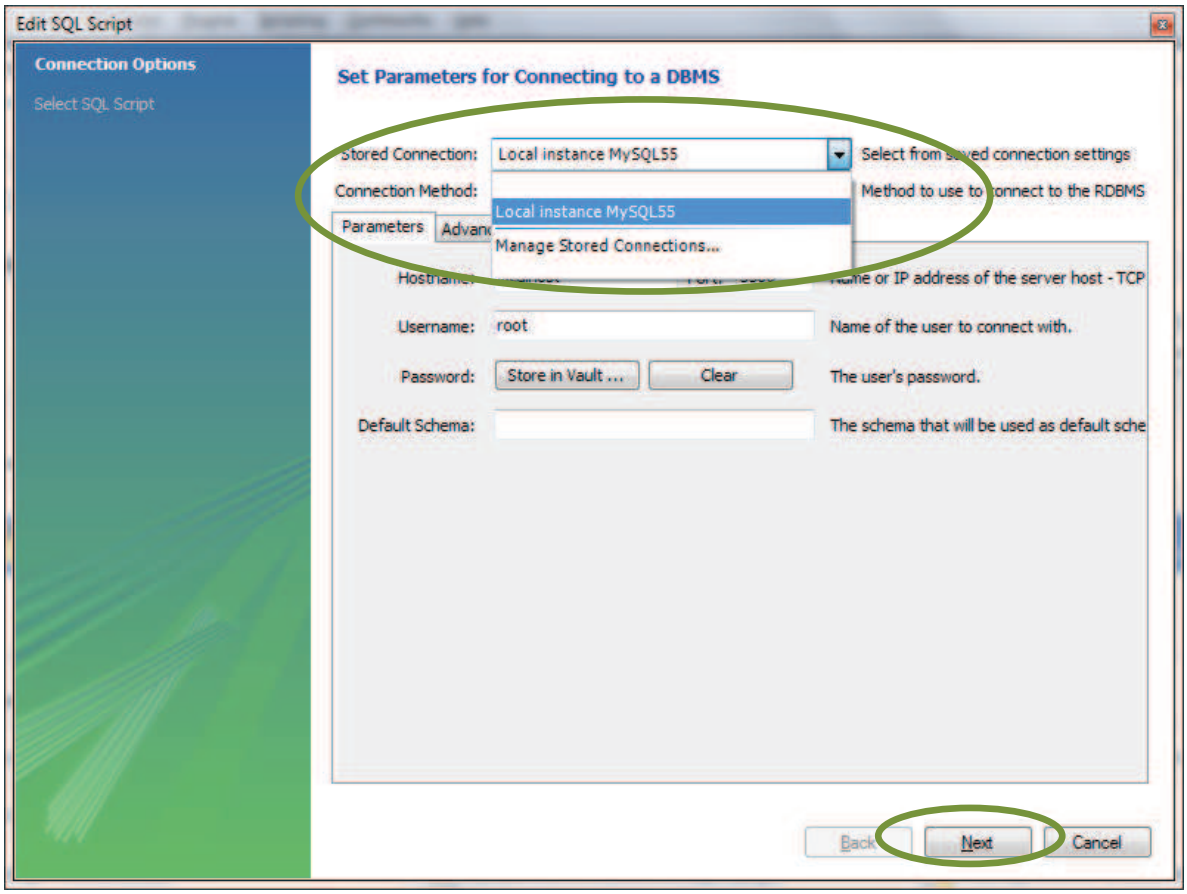




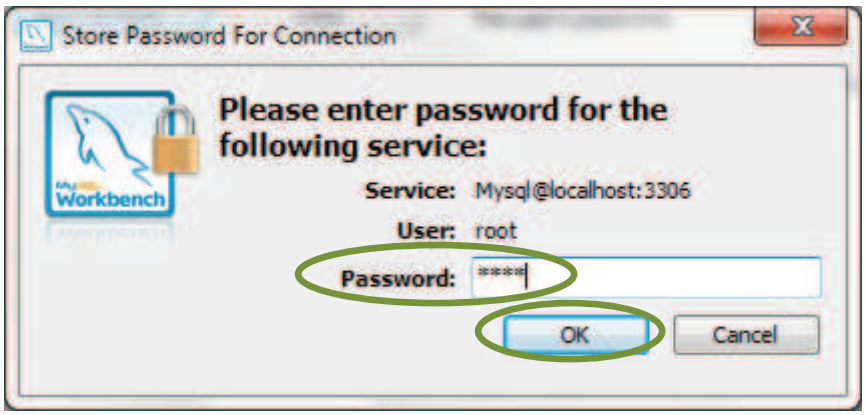
3. Una vez ya abierto el programa tenemos una pantalla de bienvenida como esta. Damos clic en la opción **Edit SQL Script**



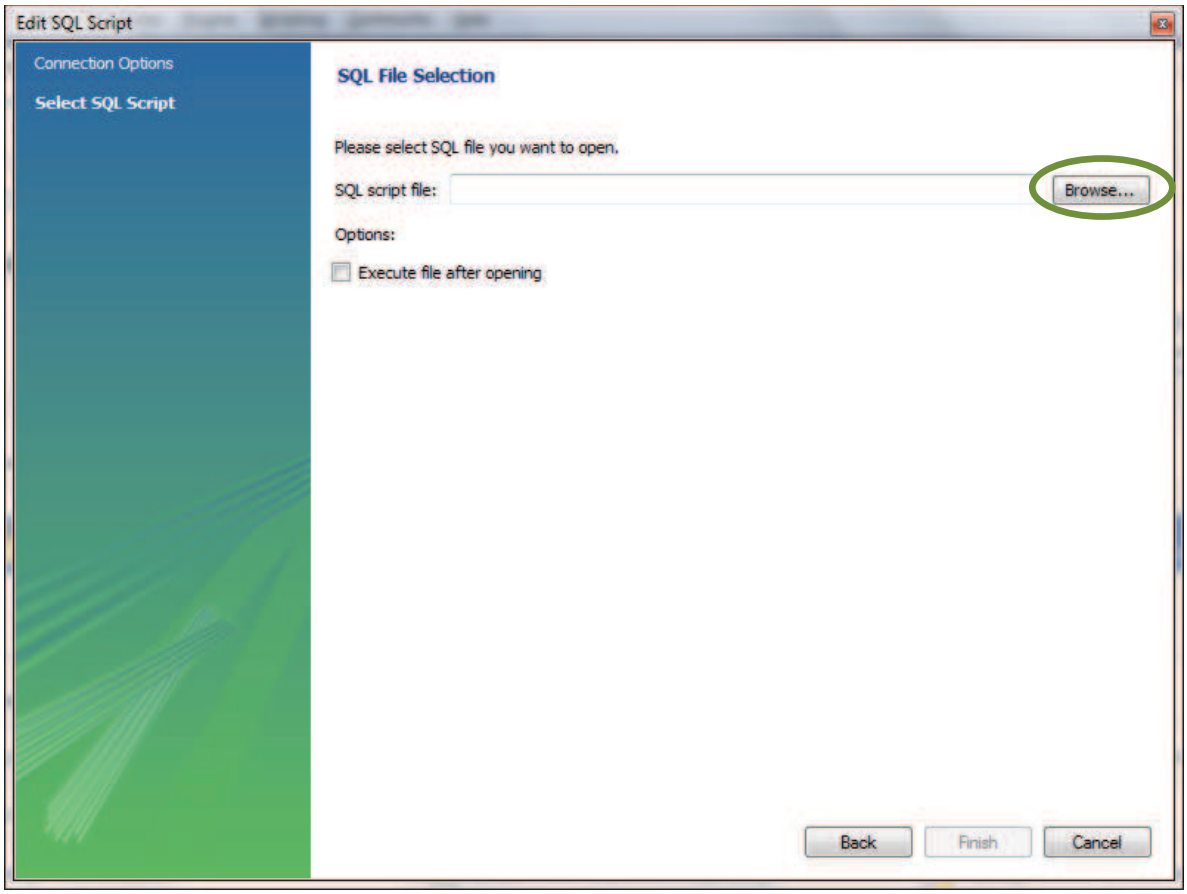
4. En esta ventana tenemos que escoger a conexión es que se va a conectar. Solo tenemos una y es la que vamos a escoger. Y damos clic en **Next**



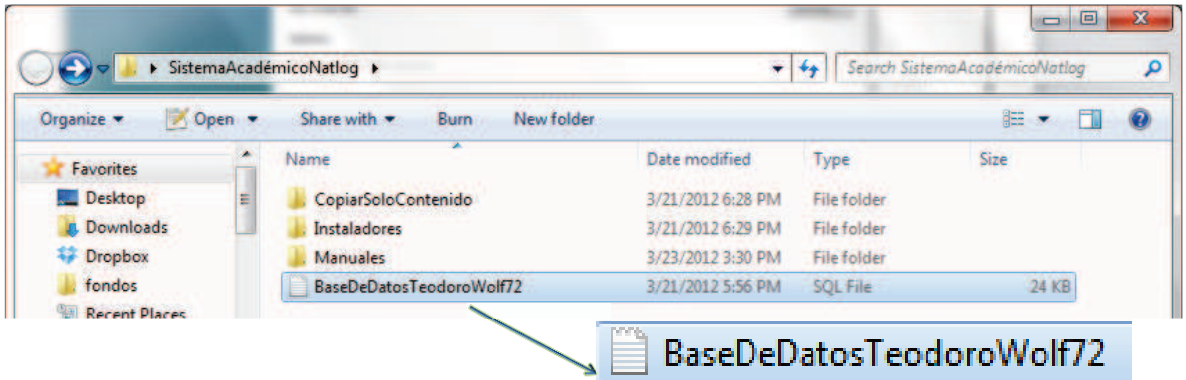
5. En caso de que nos pida la clave ponemos en el campo **Password:** "root" (sin las comillas) y damos clic en **OK**



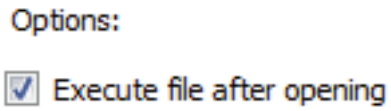
6. Procedemos a buscar el archivo proporcionado en el CD poniendo en el botón Buscar (Brownse en inglés). Se abre una ventana donde debemos navegar hasta el CD y buscar el archivo.



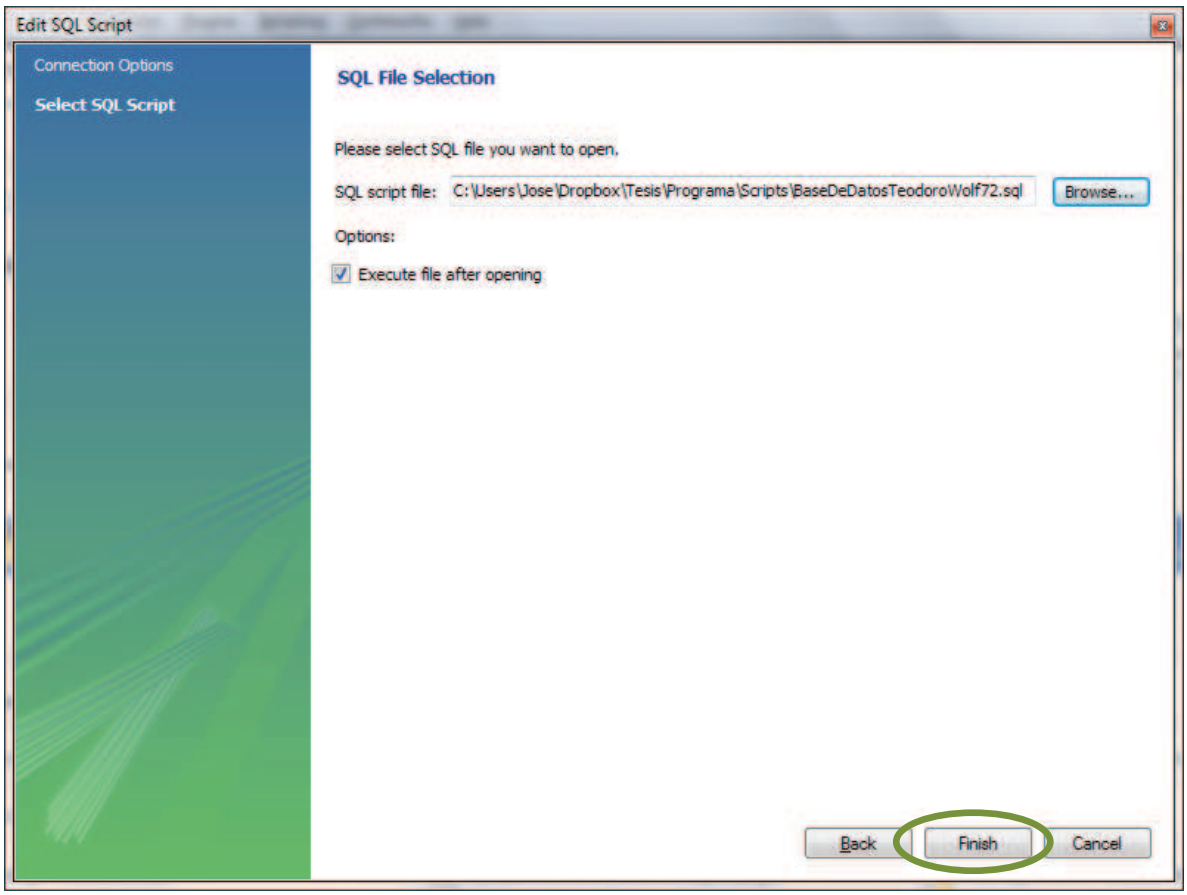
7. El archivo que debemos localizar es el siguiente BaseDeDatosTeodoroWolf72



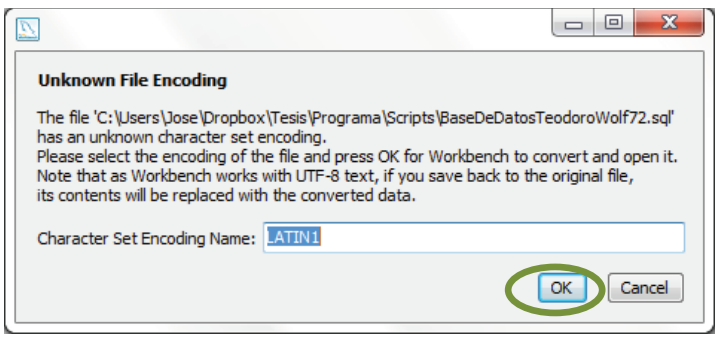
8. Y a seleccionar la casilla **Execute file after opening** (debe estar con un visto)



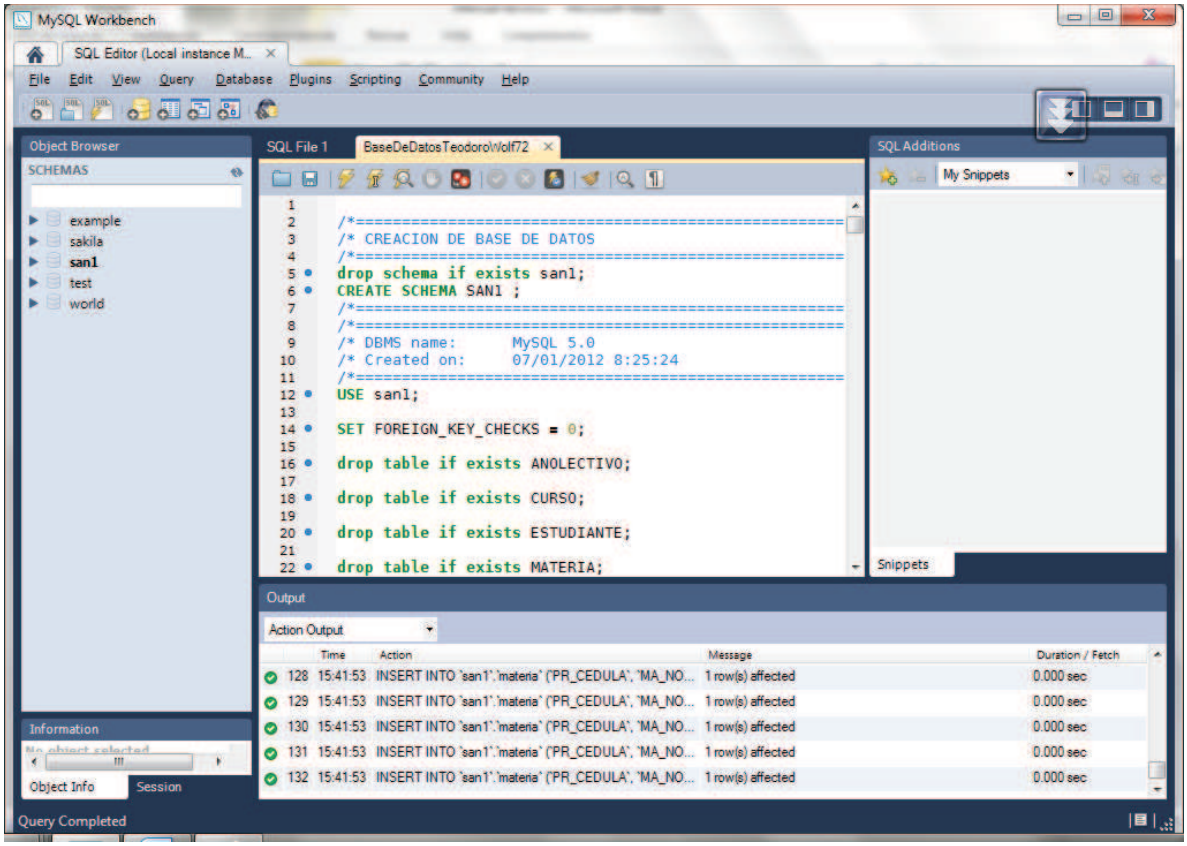
9. La ventana debe quedar de la siguiente forma. Damos clic en **Finish**



10. Si nos aparece este mensaje, solo damos clic en OK



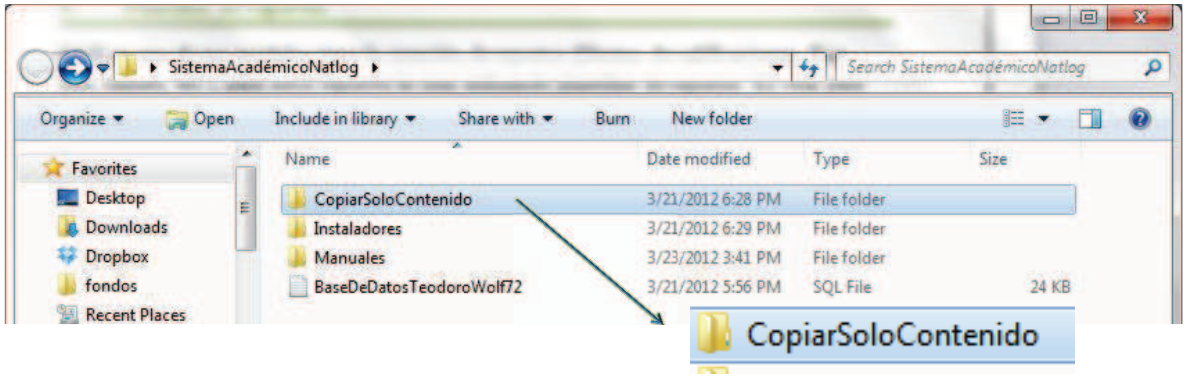
11. Nos debe aparecer una ventana como esta, donde el proceso ha culminado correctamente.



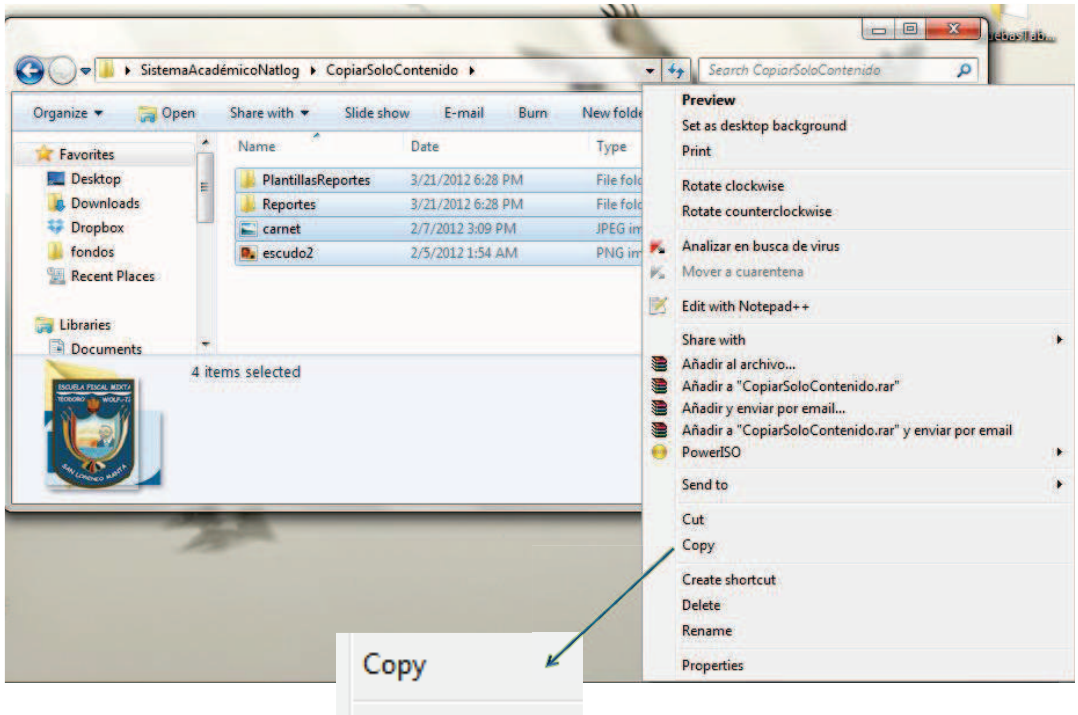
6. Plantillas de reportes.

SAN en uno de sus módulos tiene la creación de reportes (libretas de calificaciones, Kardex, carnets, etc.), para estos reportes lo crea utilizando plantillas de reportes. Es vital para que los reportes funcionen correctamente realizar los siguientes pasos:

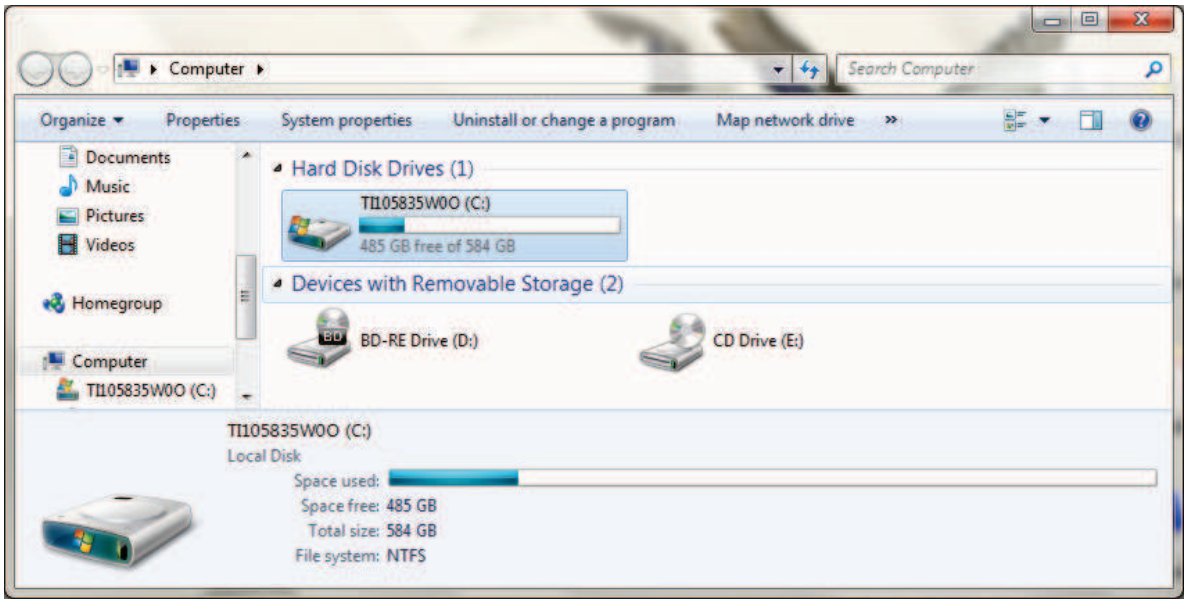
Abrimos la carpeta "CopiarSoloContenido"



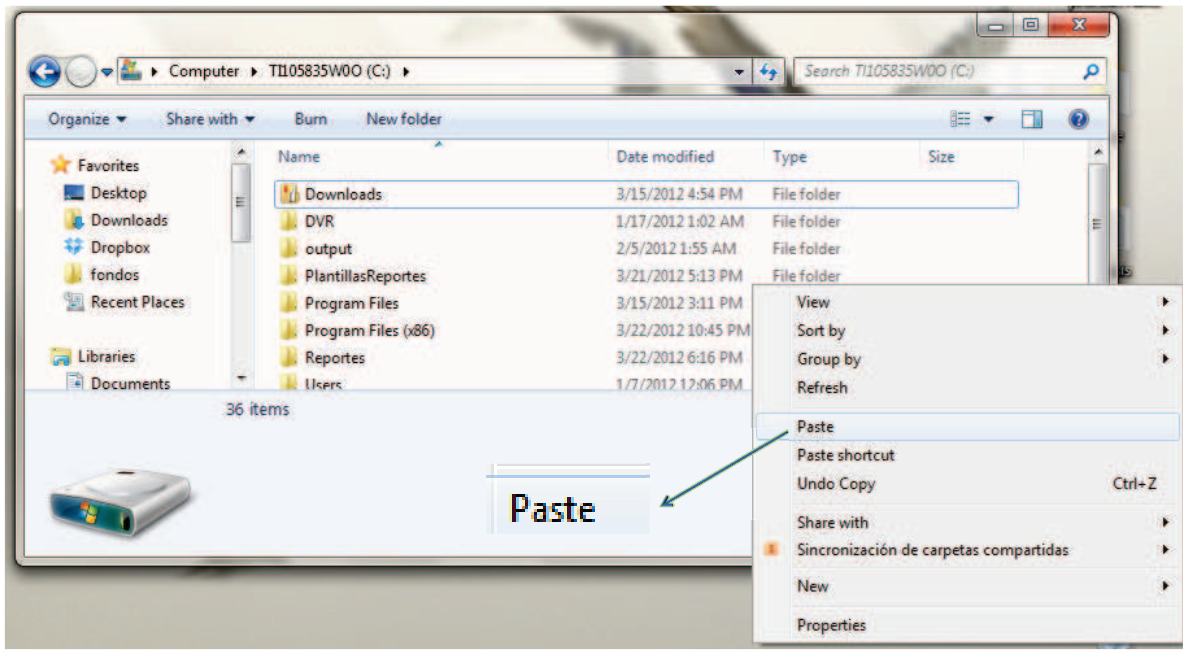
Adentro hay 2 archivos y 2 carpetas, los cuales los seleccionamos con el mouse (manteniendo pulsado el clic izquierdo) luego de esto damos clic derecho y ponemos la opción Copiar (Copy en inglés)



12. Luego nos dirigimos a MI PC y abrimos el disco duro C:



13. Damos clic derecho en cualquier pates del contenido y presionamos Pegar (Paste en inglés) Verificar que se hayan copiado en la raíz y no en alguna subcarpeta.

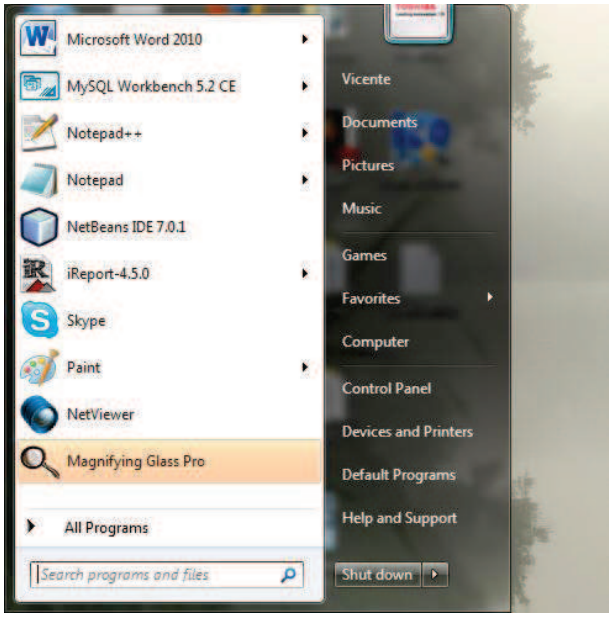


NOTA: Hay veces que se necesita permisos de Administrador para poder copiar todo esto al directorio raíz C:

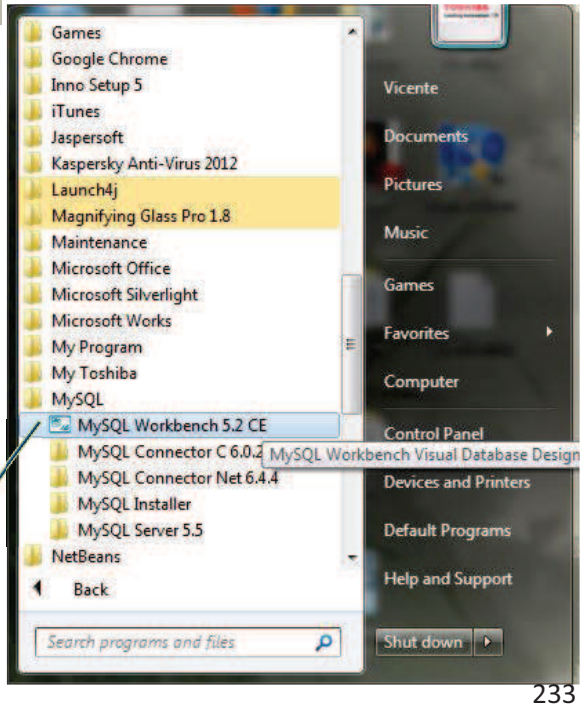
7. Cargar un respaldo de base de datos.

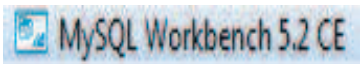
SAN en uno de sus módulos realiza respaldo de datos, y crear un archivo donde se guarda toda la información que se ha introducido en él. Este archivo no puede ser cargado o restaurado en la base de datos directamente desde SAN por motivos de seguridad. Por esta razon se lo realiza desde el motor de base de datos MySQL, para poder cargarlo se siguen lo mismo pasos del punto 5. Creación de la base de datos, con la diferencia que el archivo a buscar es el que hemos creado, se va a repetir dicha información.

1. Nos localizamos en Inicio. Luego escogemos "Todos los programas" (All Programs cuando el sistema operativo esta en inglés).

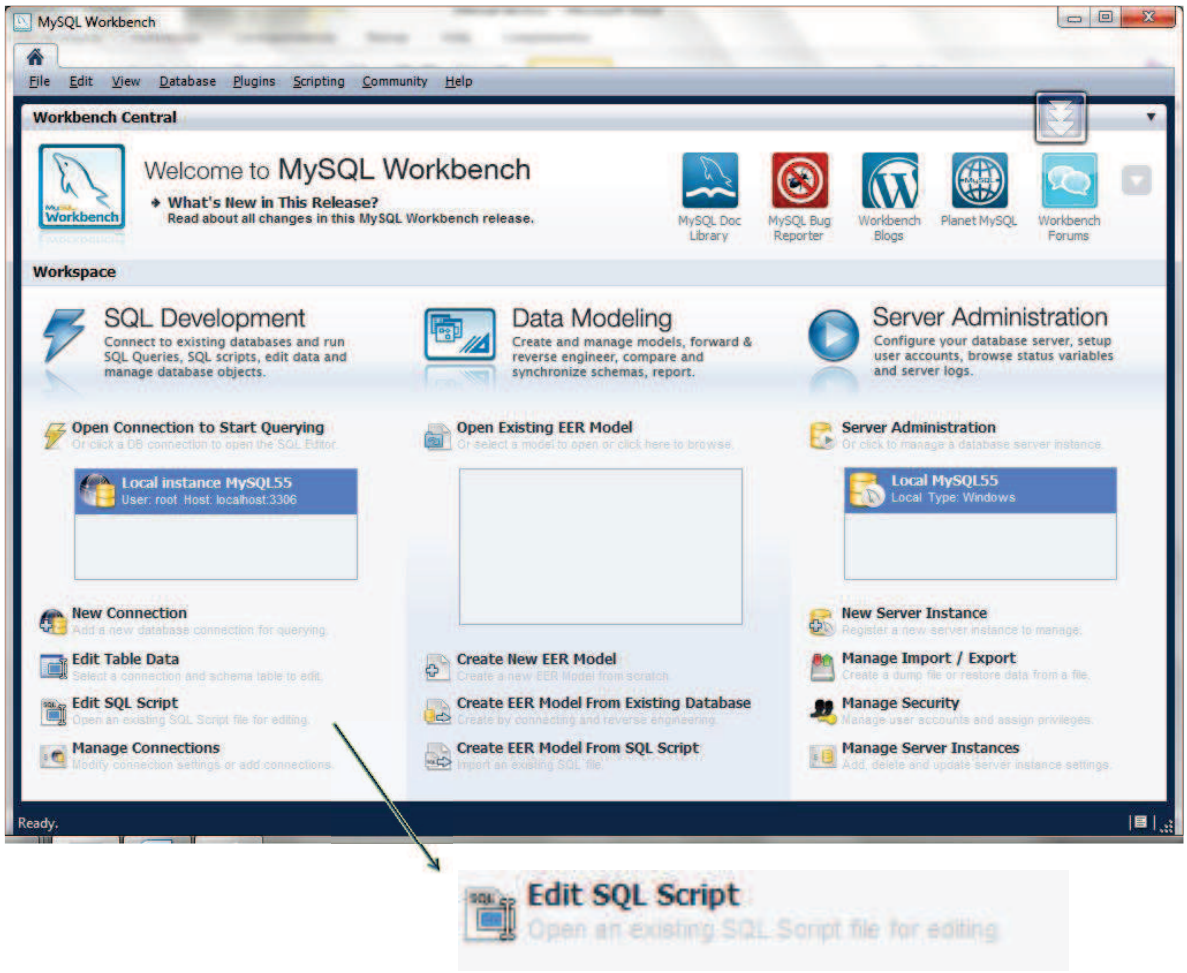


2. Buscamos la carpeta MySQL y damos clic en MySQL Workbrenach 5.2 CE

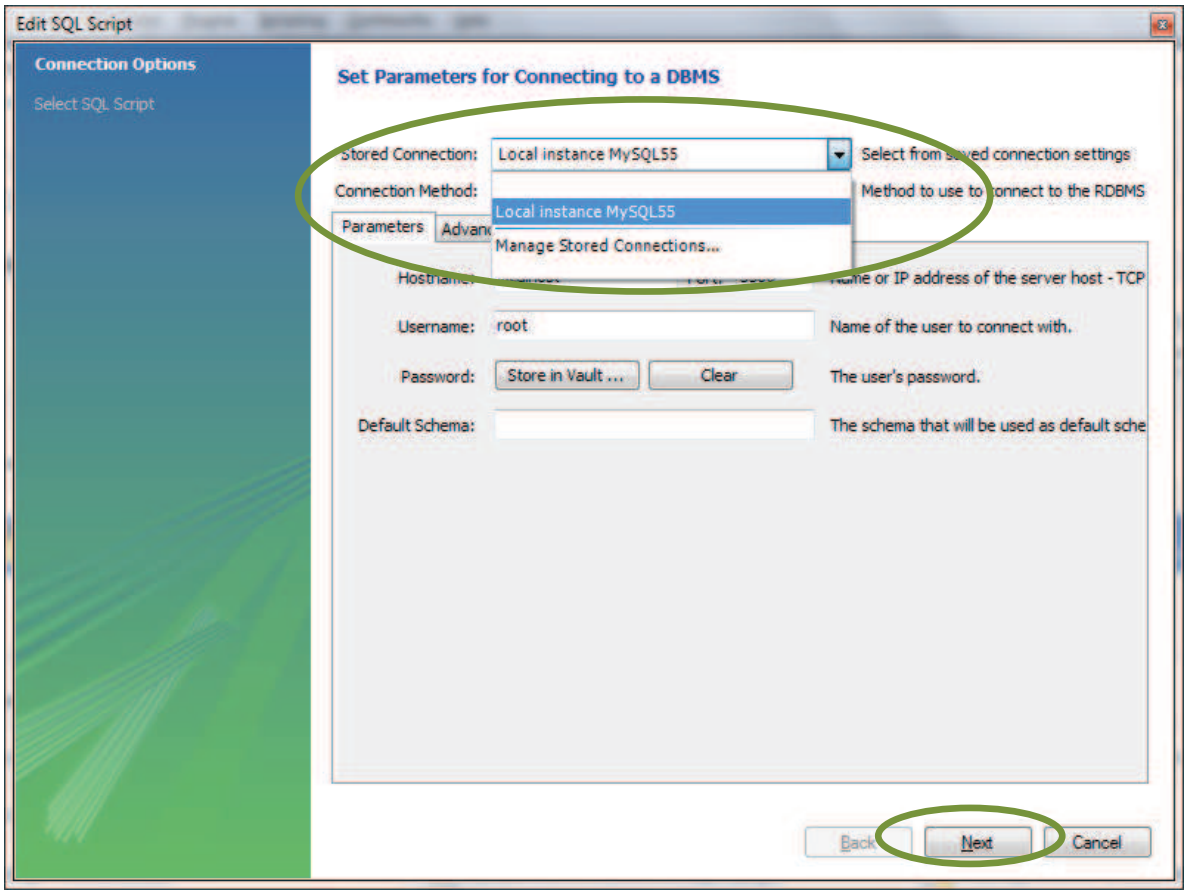


3. Una 

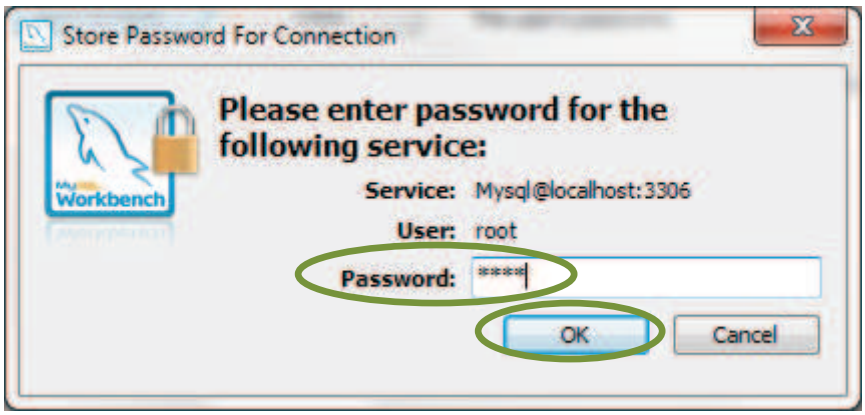
vez ya abierto el programa tenemos una pantalla de bienvenida como esta. Damos clic en la opción **Edit SQL Script**



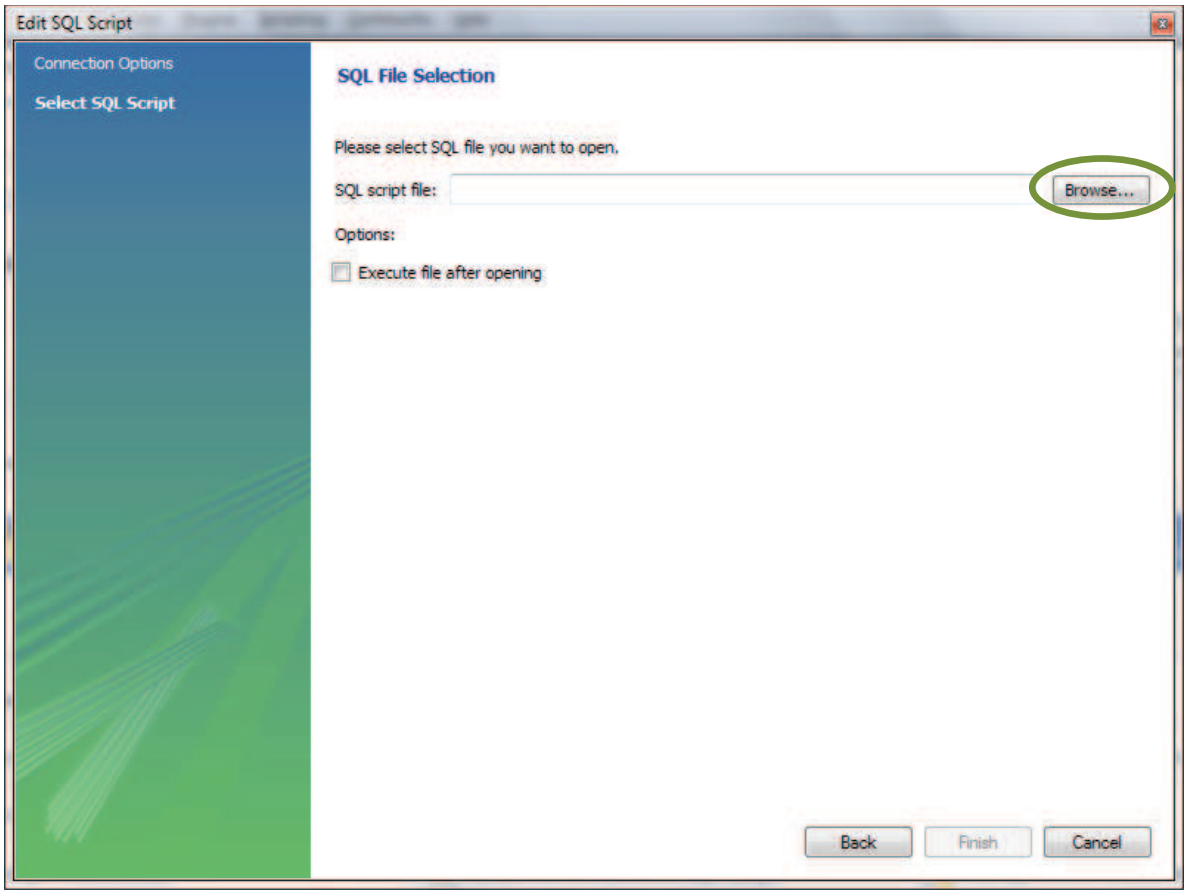
4. En esta ventana tenemos que escoger a conexión es que se va a conectar. Solo tenemos una y es la que vamos a escoger. Y damos clic en **Next**



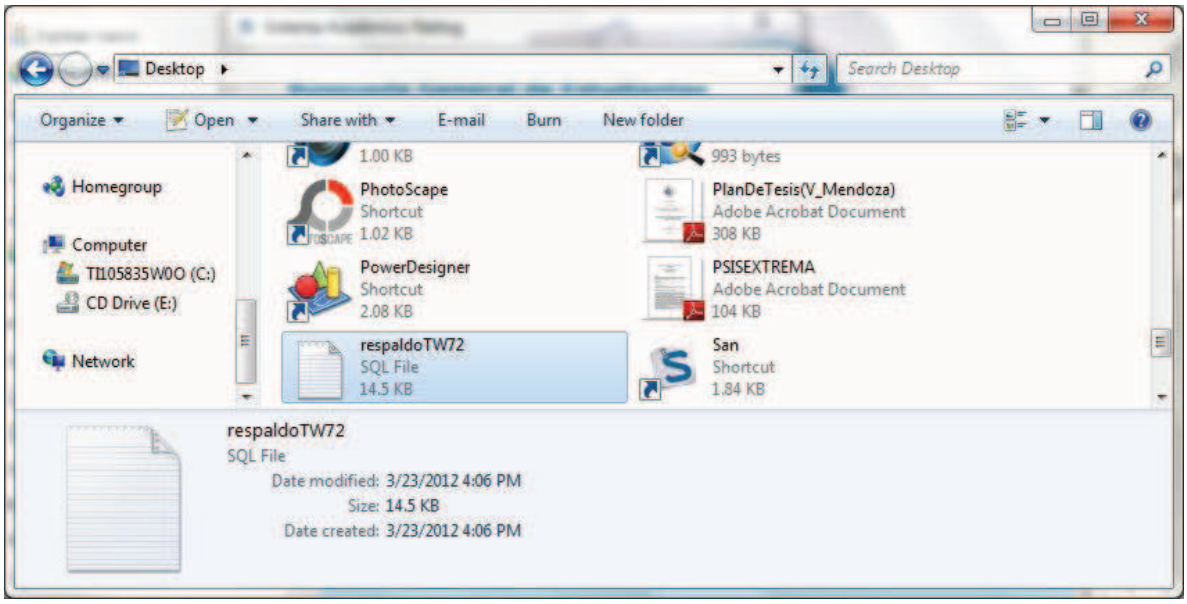
5. En caso de que nos pida la clave ponemos en el campo **Password:** "root" (sin las comillas) y damos clic en **OK**



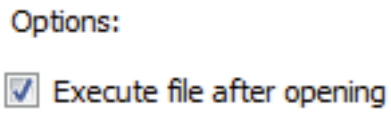
6. Procedemos a buscar el archivo que hemos creado con el programa SAN presionando el botón Buscar (Brownse en inglés). Se abre una ventana donde debemos navegar hasta el archivo de respaldo.



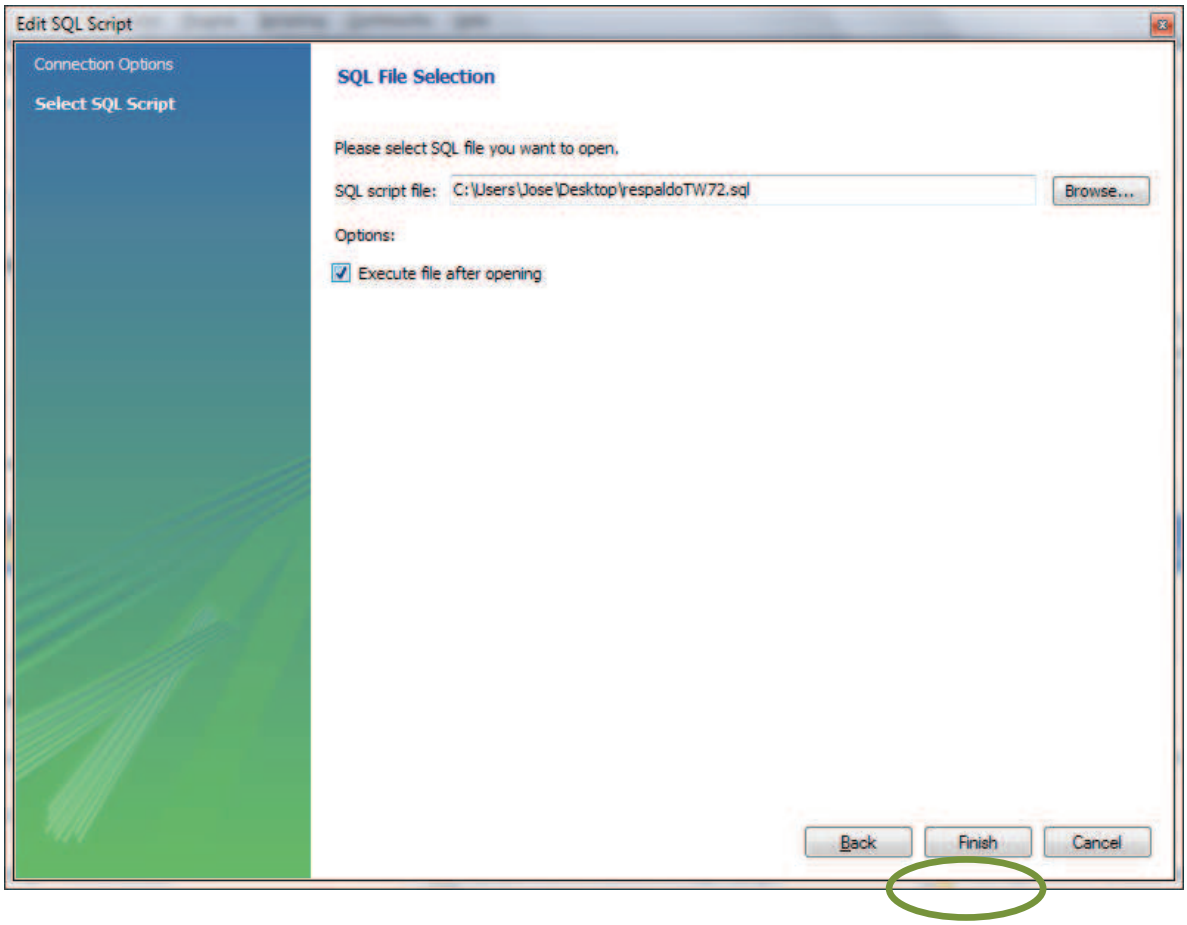
7. El archivo para efectos de este manual se llama "respaldoTW72".



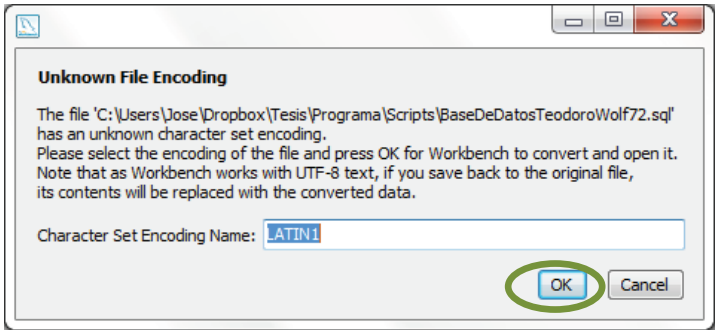
8. Y a seleccionar la casilla **Execute file after opening** (debe estar con un visto)



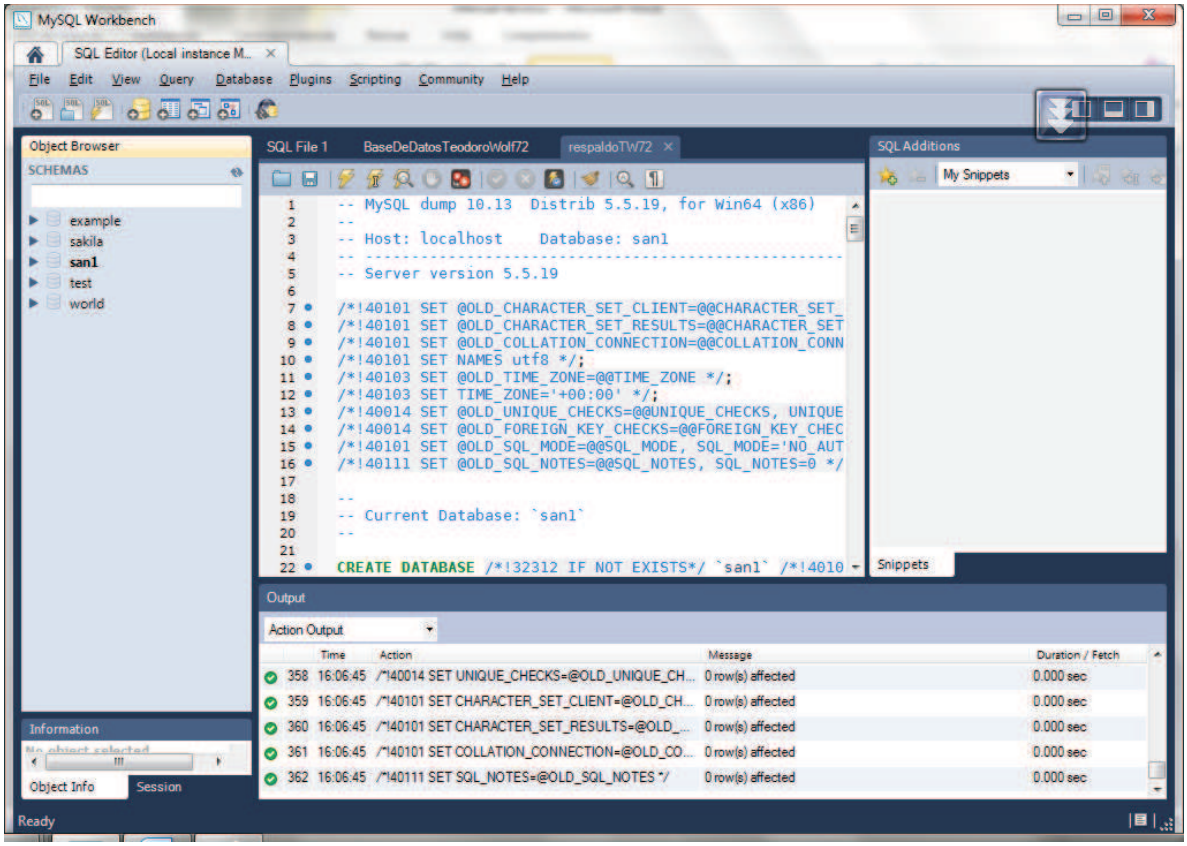
9. La ventana debe quedar de la siguiente forma. Damos clic en **Finish**



10. Si nos aparece este mensaje, solo damos clic en OK



11. Nos debe aparecer una ventana como esta, donde el proceso ha culminado correctamente.



8. Indicaciones finales.

El Sistema Académico Natlog(SAN) fue diseñado para ser lo más intuitivo posible, aun así es vital que sean los dos manuales antes de comenzar a utilizarlo.

SAN fue desarrollado como proyecto de tesis con el nombre ***DISEÑO, DESARROLLO E IMPLEMENTACION DEL SISTEMA ACADEMICO EN ESCUELA FISCAL MIXTA "TEODORO WOLF N°72" DE SAN LORENZO – MANTA – MANABÍ***

AUTOR: *Vicente Mendoza Macías*

E-Mail: ***jvmendozam@gmail.com***

Numero Celular: *092833061*