

UNIVERSIDAD AMAZÓNICA DE PANDO
ÁREA CIENCIAS Y TECNOLOGÍA
CARRERA INGENIERÍA INFORMÁTICA



INFORME FINAL DE TRABAJO DIRIGIDO

**“IMPLEMENTACIÓN DEL SISTEMA DE INFORMACIÓN ACADÉMICO-
ADMINISTRATIVO PARA LA UNIDAD EDUCATIVA BRUNO RACUA EN LA
LOCALIDAD DE PORVENIR”**

Postulante : Univ. Lucéla Graciela Escalera Mérida
Tutor : Lic. Eduardo Alberto Zubieta Copeticon
Supervisor : Ing. Tito Ponce Barco

Cobiya – Pando – Bolivia

2009

El presente capítulo describe los antecedentes y razones para realizar el proyecto en la Unidad Educativa Bruno Racua, determinando los problemas que dificultan y perjudican el normal desempeño en los procesos Académicos-Administrativos, describiendo las causas que ocasionan el problema principal y los efectos que conllevan, dando respuestas a cada uno de ellos, se detalla la solución propuesta, el objetivo general y los objetivos específicos en la institución. Además se indican los alcances y las limitaciones del presente trabajo dirigido, la metodología y las herramientas que se usara para el análisis e implementación del proyecto.

1.1. INTRODUCCIÓN

Vivimos en un mundo globalizado, donde los avances tecnológicos han tenido transformaciones en distintas áreas de trabajo. Actualmente la educación no queda al margen de estas transformaciones, la educación en Colegios a nivel internacional y nacional, implementa sistemas de información para la automatización de procesos académicos, administrativos y financieros.

Es de tal manera que desde 1994, Bolivia lleva adelante el "Programa de Reforma Educativa (PRE) en el que se establece el carácter democrático de la educación, por cuanto la sociedad participa en su planificación, organización, ejecución y evaluación. Como consecuencia a este hecho, la Reforma Educativa permite dar un paso más a estos avances tecnológicos, debido a que se van implementando sistemas de información en distintos departamentos, como ser: Tarija - Colegio San Luis (Sistema de colegios), Sucre - Colegio la Inmaculada (Sistema de gestión académica), Santa Cruz – Colegio Henry Prince (Sistema académico administrativo), y otros.

Pero no es así el caso de las Unidades Educativas dentro del departamento de Pando, específicamente en la localidad de Porvenir, puesto que no cuentan con Sistemas de Información que faciliten los procesos académicos y administrativos dentro su gestión académica.

La Unidad Educativa Bruno Racua de la localidad de Porvenir, en la actualidad realiza procesos semi automatizados (excel) en la inscripción de estudiantes, control de notas, y procesos manuales en la elaboración de libretas, surgiendo errores que no son detectados y corregidos oportunamente, desperdiándose tiempo en la búsqueda de documentos e información solicitada

por los interesados, acarreado consigo disconformidad a padres de familia y contratiempos en el desenvolvimiento normal del colegio.

Tomando en cuenta las desventajas que tiene el actual sistema semi automatizado, se plantea como propuesta de solución del presente Trabajo Dirigido, mejorar los procesos académicos y administrativos de la Unidad Educativa Bruno Racua , mediante la ***implementación del Sistema de Información Académico-Administrativo***, para apoyar de esta manera oportunamente a la toma dediciones en la gestión académica - administrativa.

1.2. PLANTEAMIENTO DEL PROBLEMA

De acuerdo al análisis previo realizado en la unidad educativa Bruno Racua, se logró rescatar información de los procesos académicos-administrativos, de los cuales se pudo identificar los siguientes problemas.

- Dificultad en el manejo de volúmenes de información respecto a los estudiantes.
- Demora en la elaboración de libretas de calificación de los estudiantes.
- Fallas en la transcripción de notas trimestrales, anuales e información de estudiantes.
- Procesos tediosos y rutinarios en la matriculación de estudiantes.
- Retardo en la organización de la información de estudiantes.
- Poca fluidez en la búsqueda de información de estudiantes.

Por todo lo mencionado con anterioridad, se define el siguiente problema primordial:

“Los procesos Académico-Administrativos son lentos con resultados poco eficaces influyendo en la toma de decisiones de la U.E.B.R. Trayendo consigo un arduo acceso a la información y a datos no centralizados”.

Como consecuencias de los problemas mencionados con anterioridad se desarrollan simultáneamente los siguientes efectos.

- Pérdida tiempo para el interesado, en los procesos académicos y administrativos.
- Incapacidad de producir el máximo los resultados, con el mínimo de recursos en energía y tiempo.
- Pérdida de tiempo en la corrección de datos.
- Mayor gasto económico en materiales de escritorio, tiempo y esfuerzo.
- Retrasos en la entrega de reportes (formularios), al SEDUCA.
- Demora en la toma de decisiones realizadas por la máxima autoridad

1.3. SOLUCIÓN PROPUESTA

Ante la situación anteriormente descrita y de acuerdo al planteamiento del problema, el presente Trabajo Dirigido tiene como propuesta de solución, la *documentación e Implementación del Sistema informático de colegios*, en la Unidad Educativa Bruno Racua, que permitirá la administración y seguimiento de la información academia-administrativa en forma adecuada y confiable.

1.4. OBJETIVOS

1.4.1 Objetivo General

Implementar el Sistema de Información Académico – Administrativo, para optimizar la gestión académica de la Unidad Educativa Bruno Racua.

1.4.2. Objetivos Específicos

- Analizar los procesos existentes, por medio del diagrama de casos de uso, para observar el funcionamiento el sistema estático.
- Analizar los procesos existentes, por medio del diagrama de secuencias, para observar el funcionamiento el sistema de forma dinámica.
- Analizar la estructura de la base de datos, aplicando la ingeniería inversa para bases de datos, documentando la Base de Datos, mediante el diseño conceptual, diseño lógico y diseño físico.
- Migrar los datos a las tablas del sistema de información académico. administrativo, para obtener los datos del anterior sistema (hojas de cálculo).
- Implementar el sistema, mediante el diagrama de despliegue, a los archivos ejecutables del sistema de información académico administrativo, en la unidad educativa Bruno RACUA.
- Realizar pruebas unitarias de caja negra al sistema para detectar fallas.
- Capacitar en el manejo del sistema a los usuarios correspondientes.

1.5. METODOLOGÍAS Y HERRAMIENTAS

La metodología a ser utilizada es la *ingeniería inversa*, que se compone de dos tipos que son:

De datos: Se aplica sobre algún código de bases datos (aplicación, código SQL, etc) para obtener los modelos relacionales o sobre el modelo relacional para obtener el diagrama entidad-relación

De lógica o de proceso: Cuando la ingeniería inversa se aplica sobre código de un programa para averiguar su lógica o sobre cualquier documento de diseño para obtener documentos de análisis o de requisitos.

La herramienta a ser utilizada es: el *Lenguaje de Modelado Unificado (UML)*, porque permite capturar el funcionamiento de un sistema para comunicarla posteriormente a quien este involucrado, esto se lleva a cabo mediante un conjunto de símbolos y diagramas, donde cada diagrama tiene fines distintos dentro de los procesos. A continuación se detalla metodologías y herramientas:

Ingeniería Inversa

ETAPA	DESCRIPCION DE LA ETAPA	HERRAMIENTA E INSTRUMENTOS	RELACIÓN ENTRE OBJETIVO Y ETAPA
Ing. Inversa de Procesos	Análisis de procesos estáticos	Diagramas de Casos de Uso, por medio de la herramienta de UML.	Representación y documentación de los procesos que intervienen en el sistema de forma estática.
	Análisis y de procesos dinámicos	Diagramas de Secuencias por medio del Lenguaje de proceso Unificado.	Representación y documentación de procesos dinámicos existentes en el sistema.
Ing. Inversa de Datos	Analizar la estructura de la base de datos	Diagramas de clases para la bases de datos.	Determinar la estructura de la base de datos.
	Extracción de información y almacenamiento en repositorios (entidades, atributos y otros).	Aplicación del diseño conceptual, diseño lógico y diseño físico	Documentación de la Base de Datos.

IMPLEMENTACION Y PRUEBAS

ETAPA	DESCRIPCION DE LA ETAPA	HERRAMIENTA E INSTRUMENTOS	RELACIÓN ENTRE OBJETIVO Y ETAPA
Implementación	Instalación del sistema Académico – Administrativo, Configuración de variables, del Servidor y gestor de bases de datos	Servidor Apache PostgreSQL.	Se obtendrán todos los componentes ejecutables necesarios para el funcionamiento del sistema.
	Implementación de la Base de Datos.	PostgreSQL	Contar con una base de datos multi-usuarios y más confiable.
	Migración de datos a la Base de Datos.	Adecuar el sistema a las nuevas necesidades de la base de datos.	Obtención de la nueva información en la base de datos para centralizar la información.
Pruebas	Pruebas de un número creciente de módulos, conexiones entre los módulos y pruebas de todo el sistema.	Realizar <i>pruebas de sistema</i> , pruebas unitarias al sistema.	Probar el correcto funcionamiento de que cada uno de los módulos por separado.
	Descripción de la calidad del software en términos de una o más de seis características básicas,	Calidad del software bajo las normas de calidad ISO 9126.Las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y	Contar con un sistema de calidad

		portatibilidad.	
--	--	-----------------	--

1.6. ALCANCES

El alcance del presente proyecto es la *documentación e implementación* del Sistema de Información Académico-Administrativo, automatizando los procesos y logrando de esta manera un manejo de información de forma eficiente, en los siguientes módulos.

Módulos documentados e implementados

- Módulos de Registros de personas.
- Módulo Accesos de usuarios.
- Módulo de Inscripciones.
- Asistencias de estudiantes.
- Módulo de Asignaciones.
- Organizaciones.
- Infraestructura.
- Reportes.
- Administración del personal.
- Migración de datos a la base de datos
- Implementación del sistema

1.7. RESULTADOS ESPERADOS

La implementación del sistema de información académico administrativo, automatizara el flujo de trabajo de la unidad educativa Bruno Racua, permitiendo minimizar el tiempo, mejorar la organización de los procesos y datos, como ser: administración de personas, administración de

horarios, administración usuarios, asignación de horarios, asignación de cursos, elaboración de informes.

- Por medio del análisis de procesos, se lograra una representación de los procesos estáticos, que intervienen en el sistema.
- A través del análisis de procesos, se consiguiera una representación de los procesos dinámicos, que intervienen en el sistema.
- Documentación de la Base de Datos, por medio del diseño conceptual, diseño lógico y diseño físico.
- Información restaurada y centralizada, por medio de la migración de datos a las tablas del sistema.
- Implementación de componentes ejecutables necesarios para el funcionamiento del sistema.
- Errores detectados y corregidos en el sistema, plan de pruebas ejecutadas.
- Usuarios capacitados y aptos para el manejo y funcionamiento del sistema logrando una administración adecuado del mismo.

1.8. ORGANIZACIÓN DEL DOCUMENTO

- **Capítulo I.** Es la parte introductoria del proyecto, antecedentes, el problema, los objetivos, la metodología y herramientas utilizadas para el logro de los objetivos.
- **Capítulo II.** Describe el marco conceptual del proyecto, dando a conocer la aplicación de sistema de información y las herramientas utilizadas para la implementación del sistema
- **Capítulo III.** Ingeniería Inversa del sistema de información Académico-Administrativo. Documentación de procesos estáticos, dinámicos y base de datos mediante diagramas de casos de uso y diagramas de secuencia, diagramas de clases y aplicación del diseño conceptual, diseño lógico y diseño físico.
- **Capítulo IV.** Implementación del sistema utilizando diagramas de despliegue, y Pruebas aplicadas al sistema, utilizando técnicas de pruebas simétricas de calidad.
- **Capítulo V.** Conclusiones y recomendaciones obtenidas al finalizar el trabajo dirigido.

Este capítulo describe toda la Fundación Teórica empleada para el desarrollo del Proyecto. De la misma manera se hace referencia a conceptos de sistemas de información académicos-administrativos. Además se describe con mayores detalles las metodologías y herramientas utilizadas para la implementación del Sistema de Información Académico – Administrativo en la Unidad Educativa Bruno Racua.

2.1. ANTECEDENTES

En nuestro País la educación boliviana esta claramente establecido, por la Ley de Reforma Educativa (ley 1565), decretada el 7 de julio de 1994. En dicha ley se atribuye al estado la capacidad reguladora y normativa de todos los contenidos concernientes al desarrollo del sistema Educativo Nacional, reglamentando la organización y funcionamiento para las Unidades Educativas de los Niveles Inicial, Primario y Secundario.

Las Unidades Educativas dependientes del Estado no cuentan con Sistemas de Información en los Procesos de gestión académica-administrativa dentro de nuestro Departamento, de tal forma es que la Unidad Educativa Bruno Racua forma parte de este sistema tradicional de administración educativa encontrándose al margen de los avances tecnológicos y los beneficios que conllevan.

La Unidad Educativa Bruno Racua, se encuentra situada a treinta kilómetros de la ciudad de Cobija en la localidad de Porvenir, fue fundado el 24 de junio de 1982. Cuenta con una infraestructura básica y adecuada, la que permite brindar el servicio de educación en el nivel Secundario. La parte administrativa en la actualidad esta conformado de la siguiente manera: una directora, una secretaria y un portero; ocho profesores forman la parte académica. Se identifica que dentro la gestión académica administrativa del establecimiento educativo Bruno Racua, los procesos son manejados de una forma lenta en el procesamiento de información Académica – Administrativa, debido a que todos los procesos de datos e información se vienen efectuando con un procesador de hojas de cálculo (Excel) y la matriculación se la realiza de forma manual (llenado de formularios RUDE), siendo poco segura y es ahí donde surgen los problemas: entrega

de informes, los docentes entregan a la secretaria un formulario llenado manualmente sobre sus datos personales (administrativo), formulación de planes de trabajo: los docentes entregan a la secretaria plan de trabajo por cada curso y materia (académico), desarrollo de certificados de calificaciones: los docentes entregan a la secretaria las calificaciones trimestrales de los estudiantes por cursos y materia, para que posteriormente la secretaria los transcriba en un formulario en Excel (académico) y procesos de registro e inscripción a estudiantes: este registro es llenado de manera manual por la secretaria en los formularios RUDE (fotocopias) por cada estudiante, en cada nueva gestión, para que a continuación sean transcritos a un formulario en Excel (académico-administrativo) para un adecuado cumplimiento de su gestión Académica Administrativa.

La educación en Colegios a nivel internacional y nacional implementa sistemas de información para la automatización de procesos académicos, administrativos y financieros. Como un ejemplo de estos sistemas de información implementados se citan los siguientes:

Institución	Proyecto	Localización
Colegio Británico Internacional	SISTAC es un sistema de gestión para colegios y establecimientos educacionales	Perú
Colegio Privado de Ciencias Alexander Fleming"	SISTAC	Perú
Colegio salesiano "Rosenthal de la puente"	SISTAC	Perú
Colegio San Luis	Sistema de colegios	Tarija-Bolivia
Colegio la Inmaculada	Sistema de gestión académica	Sucre - Bolivia
Colegio Henry Prince	Sistemas de gestión académica	Santa Cruz
Univ. Católica Boliviana	Sistema académico administrativo	CBBA

Tabla 2.1: Implementación de Sistemas de Información
Fuente: Elaboración Propia

De tal manera la Universidad Amazónica de Pando, como prueba piloto en la pasada gestión, inicio el Proyecto de ‘Apoyo a la Gestión Académica Administrativa Colegio Bruno Racua Porvenir’ Implementándose el Sistema de Información Académico-Administrativo beneficiando a todos los estudiantes de la unidad educativa “Bruno Rácuca”. Ahora con mas fuerza el proyecto se amplio a dos unidades educativas mas, como son el colegio “Héroes de la Distancia” y el colegio “Dr. Antonio Vaca Diez” del departamento de Pando en la ciudad de Cobija.

2.2 ENFOQUE AL SISTEMA DE INFORMACION ACADEMICO-ADMINISTRATIVO

El desarrollo tecnológico a provocado un crecimiento inusitado en el rendimiento de los Sistemas de Información permitiéndonos procesar una mayor cantidad de datos, en forma rápida y eficaz, a la vez nos permite organizar y transmitir la información mas completa reduciendo los niveles de incertidumbre en el proceso de toma de decisiones.

2.2.1 Sistemas de Información

Según [LAN;1998] un **sistema de información:** "Es un conjunto de procedimientos Interrelacionados que forman un todo, es decir, obtiene, procesa, almacena y distribuye información (datos manipulados) para apoyar la toma de decisiones y el control en una organización".

Por " dinámica " se desea señalar una forma activa ", no necesariamente lineal o proporcional, y adaptada a cada situación momentánea [Web 1].

En esencia Un sistema de información (SI) es un conjunto organizado de elementos, estos elementos son de 4 tipos:

- Personas.
- Datos.

- Actividades o técnicas de trabajo.
- Recursos materiales en general (típicamente recursos informáticos y de comunicación, aunque no tienen por qué ser de este tipo obligatoriamente).

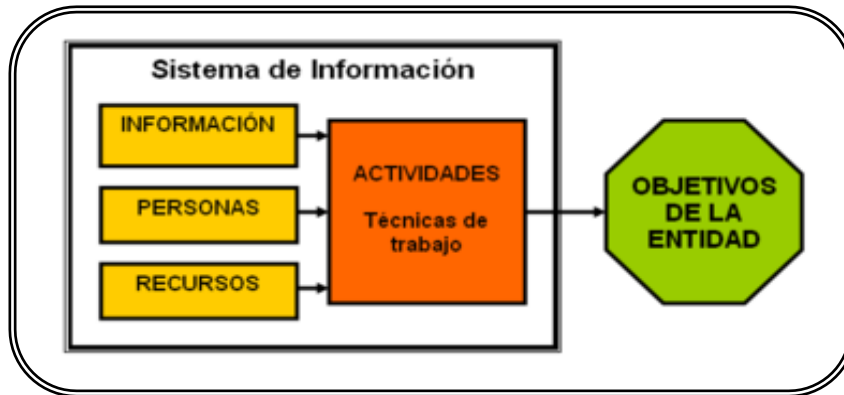


Figura 2.1: Tipo de sistemas de información
Fuente: Wikipedia en inglés, bajo licencia GFDL.

Según wikipedia, [Web 2] los sistemas de información se clasifican en:

Sistema de procesamiento de transacciones (**TPS**).

- Sistemas de información gerencial (**MIS**)
- Sistemas de soporte a decisiones (**DSS**)
- Sistemas de información ejecutiva (**EIS**)
- Sistemas de automatización de oficinas (**OAS**)
- Sistema experto (**SE**)
- Sistema Planificación de Recursos (**ERP**)

Encontrándose los **Sistemas de Información Académico-Administrativo**, dentro de los Sistemas de apoyo a la toma de dediciones, debido a la información que generan, sirve de apoyo a los mandos intermedios y de alta administración en el proceso de la toma de decisiones, Suelen ser SI interactivos y amigables, con altos estándares de diseño gráfico y visual, porque están dirigidos al usuario final.

Dentro de las Instituciones de Educación a nivel Superior se puede identificar dos tipos de procesos institucionales que con diferentes matices se desarrollan en su seno, los procesos Académicos y Administrativos, sin embargo ambos se desarrollan simultáneamente debido al necesario grado de complementariedad para el logro de objetivos organizacionales [ACR:2007].

Por lo tanto el *Sistemas de Información Académico-Administrativos* se puede definir como un conjunto de herramientas que apoyan el registro de información personal como académica de los estudiantes respondiendo a los procesos administrativos y la gestión académica de la Institución Este sistema procesa, almacena, controla y administra los diferentes datos e información académico de estudiantes e información académica-administrativa de la institución.

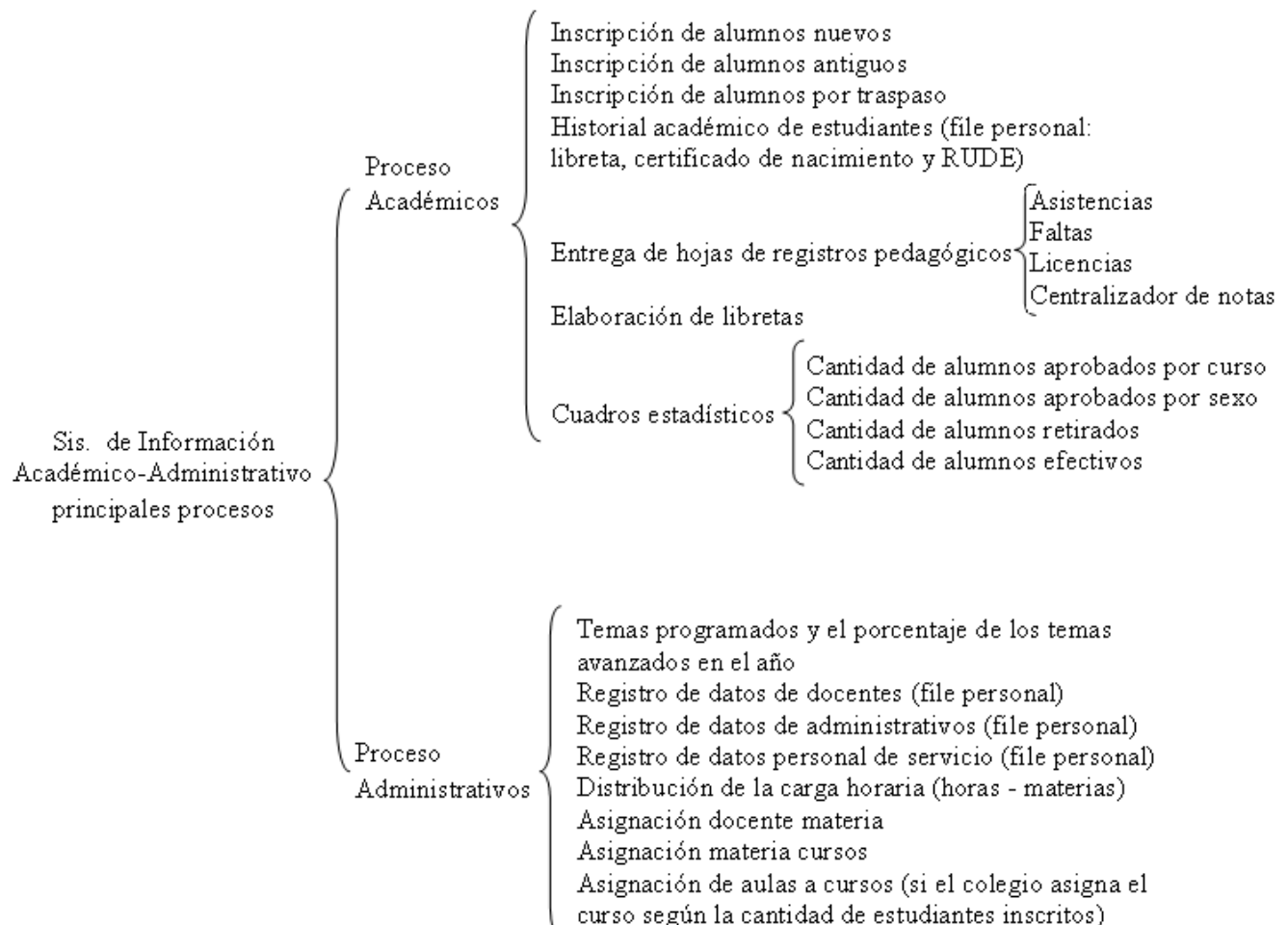


Figura 2.2: Principales procesos del Sis. de información académico-administrativo.

Fuente : Elaboración Propia

2.3 REGISTRO ÚNICO DE ESTUDIANTES (RUDE)

El Registro Único de Estudiantes [Ley 1], es un medio informático creado bajo RESOLUCION MINISTERIAL N° 001/08 del 2 de enero del 2008, del Ministerio de Educación y Culturas de la Republica de Bolivia, en la que indica lo siguiente:

1.10. Registro. Las inscripciones deben ser registradas correlativamente en el Registro Único de Estudiante (RUDE) en reemplazo del libro de inscripciones, tanto en las unidades educativas públicas y privadas, el mismo deberá estar firmado por el director o directora de la unidad educativa y refrendada por el director o directora distrital. Ningún otro tipo de registro tendrá valor legal. El director de unidad educativa, consejo de maestros y la junta escolar quedan encargados de cerrar inscripciones.

a) En caso de estudiantes nuevos deben ser inscritos en el Registro Único de Estudiantes (RUDE).

b) En caso de estudiantes antiguos, se debe regularizar su Registro Único de Estudiante de acuerdo al grado y nivel que corresponde, una vez confirmada su inscripción.

El Sistema de Información Académico-Administrativo contempla esta resolución, en la etapa de inscripción de estudiantes nuevos como antiguos.

2.4 METODOLOGIA DE DESARROLLO

2.4.1. INGENIERÍA INVERSA

Según el autor Ian.Sommerville [SI; 2005], en su libro "Ingeniería de Software", la reingeniería del software se refiere a la reimplementación de los sistemas heredados para hacerlos más entendibles. La motivación detrás de la reingeniería es conseguir un asidero para el problema creciente del mantenimiento de software. La rápida evolución de la

tecnología del hardware y el software durante los últimos años, ha dejado una herencia de millones de líneas de código propensas al fracaso, escritas en un conglomerado de lenguajes proyectándose en una mezcla de hardware incompatible. La reingeniería se define como el examen y la alteración de un sistema de software para reconstruirla y reimplementarlo en una nueva forma. El proceso de la reingeniería involucra la recuperación del diseño de una aplicación existente y el uso de esa información para reconstituirla mejorando su calidad y disminuyendo el costo de su mantenimiento. Aún cuando en la reingeniería implementar funciones existentes del sistema de una manera más eficiente, a menudo también se agregan funciones nuevas o perfeccionadas (UNC:2003).

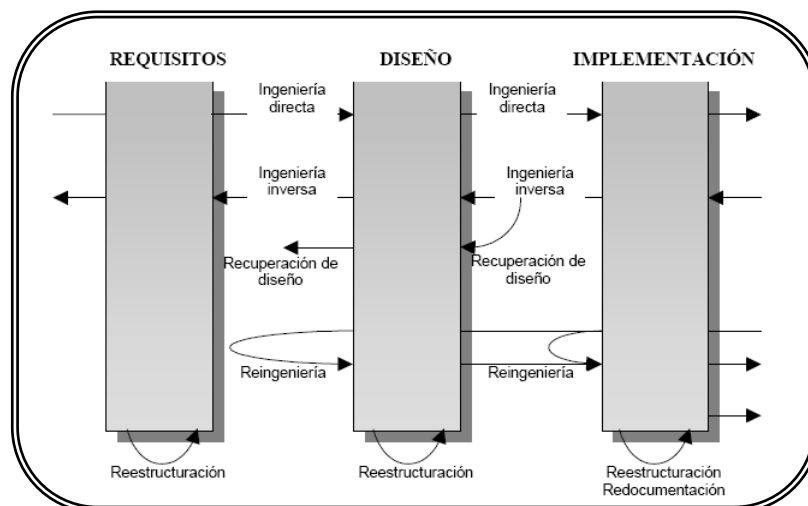


Figura 2.3: La reingeniería y sus procesos.

Fuente : Universidad Nacional de Catamarca

Consiste principalmente en recuperar el diseño de una aplicación a partir del código, los cuales se realizan mediante herramientas que extraen información de los datos, procedimientos y arquitectura del sistema existente. Es aplicable a sistema con las siguientes características:

- Tecnología obsoleta
- Documentación inexistente u obsoleta

- No especificaciones de diseño
- La aplicación cubre gran parte de los requisitos y del rendimiento esperado
- Caro de mantener
- Sólo mantenible por un número pequeño de personas

El núcleo de la ingeniería inversa es una actividad denominada extracción de abstracciones. El ingeniero tiene que evaluar el viejo programa y a partir del código fuente (que no suele estar documentado) tiene que extraer una especificación del procesamiento que se realizara, la interfaz de usuario que se aplica y las estructuras de datos o bases de datos que se utiliza (Pressman 2003).

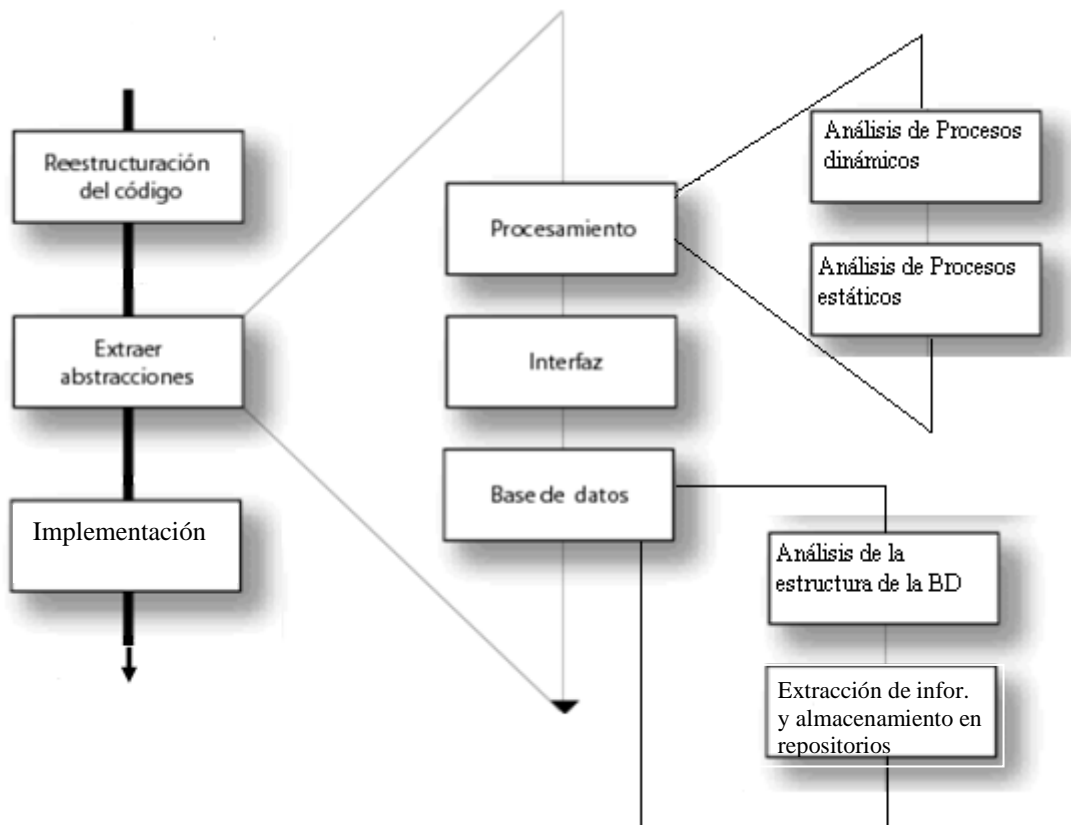


Fig.2.4: El proceso de Ingeniería Inversa
Fuente: Elaboración: Propia

El objetivo de la ingeniería inversa es obtener información a partir de un producto accesible al público, con el fin de determinar de qué está hecho, que lo hace funcionar y cómo fue fabricado.

2.4.2. RESTRUCTURACION DEL CODIGO:

Dentro de la reestructuración del código, el sistema de información académico administrativo fue desarrollado bajo normas y estándares (reforma educativa), del Servicio Departamental de Educación (SEDUCA), a nivel nacional, por lo tanto la reestructuración del código no fue necesaria, debido a que el mencionado sistema se encuentra estandarizado para las unidades educativas de nuestro estado, por otro lado, los cambios realizados en el mencionado sistema no fueron significativos y no afectaron la parte lógica del sistema.

2.4.3. EXTRACCIÓN DE ABSTRACCIONES:

2.4.3.1. PROCESAMIENTO

- a) **Análisis de Procesos Estáticos:** Dentro del análisis de los procesos estáticos se encuentran los diagramas de comportamiento, estos diagramas a su vez están divididos en tres, para este proyecto específicamente se hará uso de los **diagramas de casos de uso**. Los casos de uso describen las acciones del sistema desde el punto de vista del usuario.

- b) **Análisis de Procesos Dinámicos:** Estos procesos a su vez están conformados por los diagramas de interacción, los diagramas de interacción se sub dividen en cuatro, se hará uso específicamente del **diagrama de secuencia**. Estos diagramas muestran el flujo de información en función al tiempo.

2.4.3.2. INTERFAZ

En esta etapa del proyecto, se aplico el proceso de **interfaz** para representar la relación que se tiene entre el usuario y el sistema, aplicándose con objeto de mantener la lógica interna del programa para obtener los modelos y especificaciones que sirvieron de base para la construcción de la misma.

2.4.3.3. BASE DE DATOS

- a) **Estructura de la Base de Datos:** Dentro del análisis de la estructura de la base de datos se encuentran los diagramas de estructura, estos diagramas a su vez están divididos

en seis, para este proyecto específicamente se hará uso de los **diagramas de clases**. Estos diagramas la parte más común y más importante del diseño, son estáticos ya que no describen acción, se diseñan para mostrar todas las piezas de su solución y sus relaciones.

b) Extracción y almacenamiento de información: La extracción de las estructuras existentes actualmente en el sistema de información se realizar mediante herramientas automáticas, y el almacenamiento se realizara en repositorios (entidades, atributos y otros). Aplicación del diseño conceptual, diseño lógico y diseño físico.

2.2.4. IMPLEMENTACION

Implementación del sistema, implementación de herramientas, instalación del sistema operativo en el trabajara el sistema y configuración de variables para el buen funcionamiento del sistema.

2.4.2 INGENIERÍA INVERSA DE BASE DE DATOS

La ingeniería inversa de bases de datos [Web 3], consiste en primer lugar, en recuperar toda la información posible sobre la base de datos (conceptual, lógica y física).

El proceso de ingeniería inversa sobre las bases de tantos, es comúnmente dividido en dos bases claramente definidas.

- a) Extracción de las estructuras de datos, obteniendo como resultado el esquema lógico (Fase I).
- b) Conceptualización de las estructuras de datos, obteniendo como resultado el esquema conceptual (Fase II).

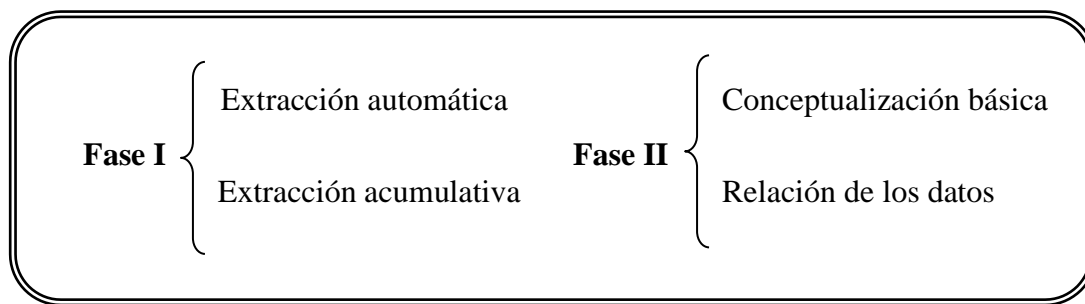


Figura 2.5: Esquema del proceso de ingeniería inversa en bases de datos

Fuente: José L. Leiva Olivencia- Universidad de Málaga (2004)

Fase I: Se realiza una extracción de las estructuras existentes actualmente en el sistema de información, dividiéndose en dos etapas de extracción de información.

Etapa 1- Extracción automática: Inicialmente se extraen mediante herramientas automáticas todas las estructuras de la base de datos como fueron diseñadas inicialmente por los desarrolladores. Se trata de una etapa típica de traducción inmediata del código para así extraer las estructuras de datos explícitas.

Etapa 2- Extracción acumulativa: Se trata de una etapa en la que la participación de los usuarios del modelo de datos con el que trabaja supondrá acumular más información de la obtenida en la etapa anterior, es posible que ciertas reglas no puedan obtenerse directamente en la etapa 1, por lo que aprovechando el conocimiento adquirido por los usuarios en el trabajo diario se podrá obtener información como ser:

- a) **Análisis de Nombres:** El usuario hará una descripción de aquellos campos en los que es posible que tengamos dudas acerca de su rol, tipos de datos, relación, etc.
- b) **Extracción de claves externas:** Sabemos que en la etapa 1, de forma sencilla se pueden obtener las claves principales, pero la obtención de claves externas a veces no es una tarea sencilla, y la información aportada por el usuario es vital.

Fase II: en esta fase se extrae el esquema conceptual a partir del esquema lógico. Denominada también como la *interpretación de las estructuras de datos*, porque se optimiza el esquema lógico.

El esquema conceptual es independiente de las estructuras físicas de almacenamiento y de la representación final de los datos que los usuarios manejan.

- 1) **Conceptualización básica:** Se describen campos, atributos, estructuras, se describe el tipo de dato y la longitud de sus campos en la bases de datos.
- 2) **Relación entre datos:** Se describen las relaciones entre datos para enlazar registros que se encuentren relacionados.

El mecanismo para obtener el esquema conceptual está basado en el propuesto por Hainaut, el cual consta fundamentalmente en la obtención de las tablas y/o entidades y sus relaciones, las tareas a ser realizadas en el esquema conceptual son:

- Identificación de entidades
- Identificación de relaciones
- Identificar atributos y asociarlos a entidades y relaciones
- Diseñar el diagrama Entidades-Relación

2.5. DICCIONARIO DE DATOS

Según [Web 4] un conjunto de meta datos (son datos que describen otros datos), que contienen las características lógicas de los datos que se utilizarán en el sistema programado, incluye nombres, descripción, alias, contenida y organización.

Estos accionarios se desarrolla durante el análisis de flujo de datos y ayuda en la determinación de requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto. En otras palabras el diccionario de datos es un catálogo, un depósito de los elementos en un sistema, identificar los procesos donde se emplea los datos y los sitios donde se necesita el acceso

inmediato a la información, el diccionario tiene dos tipos de descripciones para el flujo de datos del sistema, son los elementos datos y estructura de datos.

a) Elementos Datos: Son los bloques básicos para todos los demás datos del sistema, por sí mismos no le dan un significado suficiente al usuario, se agrupan para formar una estructura de datos. Cada entrada en el diccionario de datos es un conjunto de detalles que describen los datos utilizados por el sistema y se identifican con:

- Un nombre
- Una descripción
- Un alias
- Una longitud
- Valores de los datos

b) Estructura de Datos: Es un grupo de datos que están relacionados con otros y en conjunto describen un componente del sistema. Basados en 4 relaciones de componentes, limitándose individualmente como en conjunto con otra.

1. Relación secuencial.
2. Relación de selección.
3. Relación de interacción.
4. Relación opcional.
5. Notación.

2.6. PROGRAMACIÓN ORIENTADA A OBJETOS (OOP)

La programación orientada a objetos [Web 5], como paradigma, es una forma de pensar, una filosofía, de la cual surge una cultura nueva que incorporan técnicas y metodologías diferentes. Pero estas técnicas metodologías, y la cultura misma, provienen del paradigma, no lo hace. La OOP como paradigma es una postura Ontológica: el universo computacional está poblado por objetos, cada uno responsabilizándose por sí mismo, y comunicándose con los demás por medio de mensajes

Según Booch, los beneficios del enfoque OO son:

1. El uso del modelo OO nos ayuda a explotar el poder expresivo de todos los lenguajes de programación basados en objetos y los orientados a objetos.
2. El uso del modelo OO alienta el rechazo no sólo del software, sino de diseños completos.
3. Produce sistemas están contruidos en formas intermedias estables y por ello son más resistentes al cambio en especificaciones y tecnología.

2.7. HERRAMIENTA PARA DOCUMENTAR

2.7.1. EL LENGUAJE UNIFICADO DE MODELADO-UML

Es un lenguaje grafico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un determinado lenguaje, esquemas de Base de Datos y componentes de software rehusables es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos [Grady Booch, Ivar Jacobson y Jim Rumbaugh.: 2000].

I. Elementos del UML

Según Popkin [Web 6], son abstracciones que actúan como unidades básicas de construcción. Estos autores fueron contratados por la empresa Rational Software Co., para crear una notación unificada en la que basar la construcción de sus herramientas CASE. En el proceso de creación de UML han participado, no obstante, otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores.

a) Diagramas de Caso de Uso

Pieza de funcionalidad reusable [Web 7], que define la cadena de valor de los Actores que interactúan con un Sistema.

Para qué usamos los Casos de Uso.

1. Empaquetar los escenarios de usabilidad de un sistema.
2. Facilitar la trazabilidad entre los requerimientos y los distintos artefactos de modelado: Clases, Objetos, Acciones, Estados, etc., hasta el código.
3. Describir una secuencia de acciones que podemos organizar en escenarios tipo.
4. Representan todas las maneras posibles de usar los recursos de un sistema/organización.
5. Despiezar un proceso en funciones reutilizables.
6. Comunicar el alcance y complejidad funcional de un producto (Censo de Casos de Uso Candidatos).
7. Especificar un “contrato de servicios” entre los Actores y el Sistema.

b) Diagramas de Clases

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase) [GPJF; 2004]. A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, y otros.), en donde:

- **Superior:** Contiene el nombre de la Clase
- **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).
- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

c) Diagrama de Secuencia.

Un diagrama de Secuencia [Web 8], muestra una interacción ordenada según la secuencia temporal de eventos. En particular, muestra los objetos participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o activaciones a las que se refieren.

d) Diagramas de Despliegue.

Estos tipos de diagramas muestran al lector los elementos lógicos, sus ubicaciones físicas y como se comunican estos elementos, así como el número y variedad de elementos físicos y lógicos, los diagramas de despliegue están conformados por:

Nodos: Cajas tridimensionales que representan dispositivos físicos que pueden ser computadoras o entornos de ejecución (sistemas operativos), o entornos de auto contención (Servidores).

Artefactos: Los artefactos son las cosas que se están desplegando, los artefactos se modelan usando el símbolo de clase y un estereotipo *artefacto*, los artefactos pueden ser: EXE, DLL, archivos HTML, documentos, etc.

f) Actores

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

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

2.8. HERRAMIENTAS DE SOFTWARE

Las herramientas de Software, hacen referencia a la arquitectura en la que está desarrollada el sistema informático, la razón por la cual se realizó el proyecto en dos capas, es porque la primera capa es la interfaz gráfica de usuario, esta interfaz sirve para manipular datos de entrada y de salida, la segunda capa es la base de datos relacional, creada en PostgreSQL, por lo que no se necesita un modelo de objetos de negocios, debido a que el sistema presenta una arquitectura simple pero de gran ayuda para el apoyo en la toma de decisiones en la Unidad Educativa.

Una Base de Datos, en esencia no es más que un “sistema para archivar en computador”. La base de datos en sí puede considerarse en como una especie de archivero electrónico, dicho de otra manera, de su lugar donde se almacena un conjunto de archivos de datos computarizados, al usuario el sistema se le brindará recursos para realizar diversas operaciones sobre estos archivos. [AAI&AMF; 2008]

2.8.1. POSTGRESQL

El lenguaje SQL se ha establecido claramente como el lenguaje estándar de Base de Datos relacionales, es considerado como un lenguaje de consulta, que contiene muchas otras capacidades además de las consultas en base de datos, incluye características para definir la estructura de los datos su modificación en la BD y para las especificaciones de ligadura de seguridad [MGG,OPM; 2002].

PostgreSQL es un servidor de base de datos relacional orientada a objetos de software libre ubicado bajo licencia BSD (es la licencia de software otorgada principalmente para los sistemas BSD (*Berkeley Software Distribution*)). Pertenece al grupo de licencias de software Libre.) [7].

PostgreSQL permite que mientras un proceso escribe en una tabla, otros acceden a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit.

Entre sus características tenemos:

- Claves ajenas también denominada claves ajenas o llaves foráneas.
- Disparadores (triggers)
- Vistas
- Integridad transaccional
- Herencia de tabla
- Tipo de datos operaciones geométricas

Ventajas de PostgreSQL

- **Sus características:** PostgreSQL tiene mucha de las funcionalidades de las bases de datos comerciales.
- **Funcionamiento:** PostgreSQL se ejecuta en dos modos. El modo normal fsync, que guardan el disco toda transacción completada, es más lento que la mayoría de las bases de datos comerciales, en parte porque son pocas las que hacen esto. El modo no- fsync, menos fiables que el anterior, es más rápido que las bases de datos comerciales, aunque en este modo un crash del sistema puede significar la pérdida de alguna transacción.
- **Precio:** es gratuito, tanto para fines no comerciales como hará fines comerciales. Lo mejor es que su futuro no está sujeto a las decisiones de ninguna empresa a
- **SopORTE:** Existe abundante documentación, lista de correo, acceso director desarrolladores, empresas que ofrece consultoría, y lo más importante es que el código fuente es abierto y libre para realizar cambios.

2.8.2. HYPERTEXT PREPROCESSOR – PHP

Según [Web 9], PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas. Según Saether PHP: “Es un lenguaje de "código abierto" interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor”.

El código PHP no se mezcla con las etiquetas HTML, ya que está acotado siempre por los símbolos `<? y ?>`, de forma que para los programas compositores de páginas Web son etiquetas que no soporta y las deja tal y como están. De esta forma, puede trabajarse a la vez en el diseño visual de la página y en la funcionalidad (programación) de la misma cómodamente. PHP se ejecuta en el servidor, por eso nos permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos.

El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página XML. Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

Ventajas

- Es un lenguaje multiplataforma.
- Similar en sintaxis a C y a PERL
- Se puede incrustar código PHP con etiquetas HTML.
- El análisis léxico para recoger las variables que se pasan en la dirección lo hace PHP de forma automática.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos.

- PHP viene equipado con un conjunto de funciones de seguridad que previenen la inserción de órdenes dentro de una solicitud de datos.
- Todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Al ser libre, tiene una gran comunidad de desarrolladores:
- Los fallos de funcionamiento se localizan y reparan fácilmente.
- Permite las técnicas de Programación Orientada a Objetos, clases y herencia.
- Biblioteca nativa de funciones sumamente amplia reduciendo los tiempos de desarrollo.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

2.8.3. SMARTY

Según [Web 10], SMARTY: "Es un motor de plantillas para PHP. Más específicamente, esta herramienta facilita la manera de separar la aplicación lógica (código PHP) y el contenido en la presentación (HTML/CSS) La mejor descripción esta en una situación donde la aplicación del programador y la plantilla del diseñador juegan diferentes roles.

Un único aspecto acerca de Smarty es la compilación de la plantilla. De esta manera Smarty lee la plantilla y crea los scripts de PHP. Una vez creados, son ejecutados sobre él. Por consiguiente no existe ningún costo por analizar gramaticalmente cada archivo de template por cada requisición, y cada template puede llevar toda la ventaja del compilador de cache de PHP tal como Zend Accelerator (<http://www.zend.com/>) o PHP Accelerator (<http://www.php-accelerator.co.uk>). Los archivos de plantilla TPL describen cómo se introduce la información y se muestra en la pantalla vía una ventana de plantilla. Todos los archivos de plantilla se almacenan en el subdirectorío TPL".

Algunas de las características de SMARTY que menciona Monte Ohrt son:

- Es rápido.
- Es eficiente ya que puede interpretar el trabajo más sucio.
- No analiza gramaticalmente desde arriba el template, solo compila una vez.
- El esta atento para solo recompilar los archivos de plantilla que fueron cambiados.
- Usted puede crear funciones habituales y modificadores de variables customizados, de modo que el lenguaje de la platilla es altamente extensible.
- Sintaxis de etiquetas delimitadoras para configuración de la plantilla, así lo puede usar {}, {{}}, <!--{}-->, etc.
- Los constructores if/elseif/else/endif son pasados por el interpretador de PHP, así la sintaxis de la expresión {if ...} puede ser compleja o simple de la forma que usted quiera.
- Permite un anidamiento ilimitado de sections, ifs, etc.
- Es posible incrustar directamente código PHP en los archivos de plantilla, aunque esto puede no ser necesario (no recomendado) dado que la herramienta se puede ajustar.
- Soporte de caching incrustado
- Fuentes de Plantilla absoluto.
- Funciones habituales de manipulación de cache.

2.8.4. HTML

Según [Web 11], el acrónimo inglés de *HyperText Markup Language*, que se traduce al español como lenguaje de marcas Hipertextuales. Es un lenguaje diseñado para estructurar textos y presentar los en forma de hipertexto, que es el formato estándar de las páginas web. Eso no los formatos más populares y fáciles de aprender que existen para elaboración de documentos web.

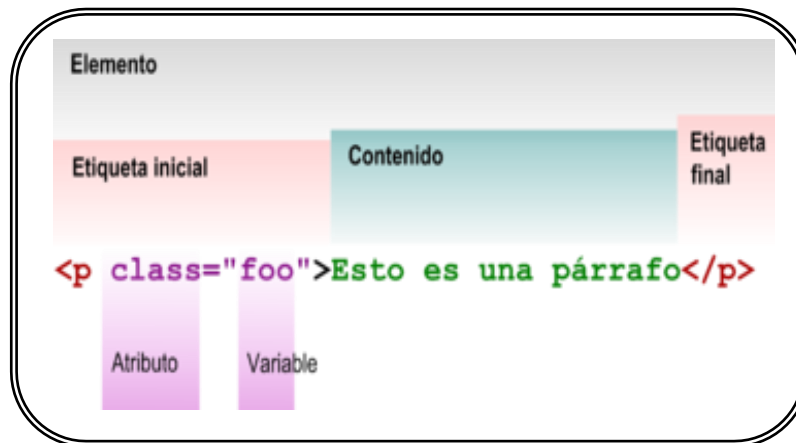


Figura2.6: Estructura general de una línea de código en el lenguaje de etiquetas HTML
Fuente: wikipedia

HTML consiste de varios componentes vitales, incluyendo *elementos* y sus *atributos*, *tipos de data*, y la *declaración de tipo de documento*.

- Los elementos son la estructura básica de HTML
- La mayoría de los atributos de un elemento son pares nombre-valor, separados por un signo de igual "=" y escritos en la etiqueta de comienzo de un elemento
- El marcado estructural describe el propósito del texto como un encabezamiento de segundo nivel.
- El marcado presentacional describe la apariencia del texto, sin importar su función

Cuatro Normas Fundamentales

a) HTML es simplemente texto

Lo primero es saber que un documento HTML es un archivo de texto simple, luego se puede editar con cualquier editor de texto sencillo.

b) Igualdad de mayúsculas y minúsculas

HTML no distingue entre mayúsculas y minúsculas en la especificación de marcas y sus atributos. Sin embargo, por legibilidad, es aconsejable codificar tanto marcas como atributos en mayúsculas.

c) No importa los tabuladores, ni los saltos de línea

Los visualizadores no toman en cuenta las tabulaciones, los saltos de línea ni los espacios en blanco extra, esto tiene ventajas y desventajas. La principal ventaja es que permite obtener resultados uniformes y de buena presentación de manera bastante fácil y la gran desventaja es que un documento HTML, por lo menos debe usar las marcas `<P>...</p>` o `
` para avitar que quede todo el texto en una sola línea.

d) Caracteres especiales

En HTML existen algunos caracteres que son especiales por que juegan un papel dentro del mecanismo del funcionamiento de HTML, como sucede con los símbolos mayor que (`>`) y menor que (`<`), y otros porque en los primeros tiempos de HTML, no formaban parte del juego de caracteres internacionales del alfabeto como sucede con los acentos.

2.8.5. APACHE TOMCAT

Según [Web 12], **Tomcat** (también llamado **Jakarta Tomcat** o **Apache Tomcat**) funciona como un contenedor de servlets (Los **servlets** son objetos que corren dentro del contexto de un contenedor de servlets (ej: Tomcat) y extienden su funcionalidad) desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems. Según Jorba ; Suppi apache es uno de los servidores más populares y de altas prestaciones de HTTP (*HyperText Transfer Protocol*). Apache tiene un diseño modular y soporta extensiones dinámicas de módulos durante su ejecución. Es altamente configurable en el número de servidores y de módulos disponibles y soporta diversos mecanismos de autenticación, control de acceso, *metafiles*, *proxy caching*, servidores virtuales, etc.

Apache está diseñado para ejecutarse como un proceso *daemon* standalone. En esta forma crea un conjunto de procesos hijo que manejarán las peticiones de entrada. También puede ejecutarse como Internet *daemon* a través de *inetd*, por lo cual se pondrá en marcha cada vez que se reciba una petición.

2.9. TÉCNICAS DE PRUEBAS DEL SOFTWARE

Las pruebas del software son un elemento crítico para la garantía de calidad del software y presentan una revisión final de las especificaciones del software las pruebas un proceso de ejecución de un programa con intención de descubrir errores en el sistema, por lo tanto las pruebas sirven para demostrar hasta qué punto el software se ajusta a las especificaciones funcionales y a los requerimientos del rendimiento establecido si inicialmente como ser:

- inspección manual del código.
- Prueba o ensayo: ejecutar y ver resultados.
- Caso de prueba: ensayo individual

Objetivos de las pruebas

- Encontrar defectos en el software.
- Una prueba tiene éxito se descubre un defecto.
- Una prueba fracasa si hay defectos y no los descubre

1. Pruebas Unitarias

Las pruebas unitarias, es una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado. El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las

partes individuales son correctas. Proporcionan un contrato escrito que el trozo de código debe satisfacer. Estas pruebas aisladas proporcionan cinco ventajas básicas:

1. **Fomentan el cambio:** Las pruebas unitarias facilitan que el programador cambie el código para mejorar su estructura (lo que se ha dado en llamar refactorización), puesto que permiten hacer pruebas sobre los cambios y así asegurarse de que los nuevos cambios no han introducido errores.
2. **Simplifica la integración:** Puesto que permiten llegar a la fase de integración con un grado alto de seguridad de que el código está funcionando correctamente. De esta manera se facilitan las pruebas de integración.
3. **Documenta el código:** Las propias pruebas son documentación del código puesto que ahí se puede ver cómo utilizarlo.
4. **Separación de la interfaz y la implementación:** Dado que la única interacción entre los casos de prueba y las unidades bajo prueba son las interfaces de estas últimas, se puede cambiar cualquiera de los dos sin afectar al otro, a veces usando objetos mock (mock object) para simular el comportamiento de objetos complejos.
5. **Los errores están más acotados y son más fáciles de localizar:** dado que tenemos pruebas unitarias que pueden desenmascararlos.

2.10. FACTORES Y MÉTRICAS PARA LA CALIDAD DEL SOFTWARE

De acuerdo con [Web 13] ISO, bajo la norma de calidad ISO-9126, ha establecido un estándar internacional para la evaluación de la calidad de productos de software el cual fue producto de software publicado en 1992 con el nombre de "Information technology - software product evaluation", donde se establecen las características de calidad para productos software.

Es una medida de factores que varían a través de diferentes aplicaciones dependiendo del requerimiento de los ya porque usuarios. Tiene por objetivo producido sistema de alta calidad donde se aplican métodos afectivos, se identifican los factores de calidad del software y se describen las actividades necesarias para conseguir los. Estas métricas se basan en el estándar

ISO/IEC 9126. El mismo que ha sido desarrollado en un intento de identificar los atributos clave de la calidad para el software, el mismo que identifica seis atributos claves de calidad [PRS ;2007].

- a) **Funcionalidad:** Grado en el que el software satisface las necesidades de los siguientes sub atributos: Corrección, Interoperatividad, Conformidad y Seguridad.
- b) **Usabilidad:** Se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser activo para el usuario en condición específica de uso.
- c) **Facilidad de Mantenimiento:** Es la facilidad con que una modificación puede ser realizada, está indicada por los sub atributos: facilidad de análisis, facilidad de cambio, estabilidad y facilidad de prueba.
- d) **Portabilidad:** Es la facilidad con la que el software puede ser llevado de un entorno a otro, los sub atributos son: facilidad de instalación, facilidad de ajustes, facilidad de adaptación al cambio.
- e) **Confiabilidad:** Es la cantidad de tiempo que el software está disponible para su uso, referido a tolerancia a fallos y facilidad de recuperación
- f) **Eficiencia:** Es el grado en que el software hace óptimo el uso de los recursos del sistema sub atributos son: tiempo de comportamiento y herramientas utilizadas.

Para la evaluación de un atributo de calidad, en el presente trabajo dirigido se hará uso de la siguiente formula:

$$a_1 = \frac{r_1 * b_1 + r_2 * b_2 + \dots + r_j * b_j + \dots + r_m * b_m}{r_1 * r_2 + \dots + r_j + \dots + r_m}$$

Figura2.7: Formula elaborada por la Universidad de Valladolid
Fuente: Universidad de Valladolid Departamento de Informática

Donde:

Sea n el número de atributos que se evalúan en la calidad de un producto

α_i Atributo a ser evaluado, donde $i = 1, n$.

r_j Representa el grado de importancia del subatributo j en la evaluación de muchas veces llamada *peso*, donde $j = 1, b$

b Es un numero entero que presenta la cantidad de subatributos que se consideran en la evaluación de α_i . La cantidad de subatributos que se consideran para la evaluación de α_i , no es constante, varia en función de su pertinencia.

b_j Es la medición de nivel presente del subatributo en la evaluación de α_i . El nivel esta establecido en la escala de 1 a 5

Este capítulo está dedicado a describir los resultados obtenidos en la implementación del Sistema de Información Académico-Administrativo, previo análisis de los procesos, aplicando las técnicas de la metodología de Ing. Inversa, y la herramienta del UML, las cuales permitirán realizar la documentación del sistema.

3.1. INTRODUCCION

Considerando la existencia del sistema para colegios y encontrándose conforme a los requerimientos de los mismos, el proyecto realizara el proceso de análisis y documentación del sistema de información académico-administrativo, para que de esta manera se permita modelar los procesos, las interfaces y la base de datos. De acuerdo a la estructura del sistema de información académico-administrativo y en relación a la aplicabilidad de la Metodología de Ing. Inversa se consideran las siguientes actividades: procesamiento (análisis de procesos), interfaz (obtención de modelos y especificaciones) y Base de Datos (análisis de la estructura, extracción de información).

Dentro del análisis de procesos se utilizan los diagramas de casos de uso, este es un diagrama de comportamiento, el cual reflejara claramente los procedimientos de forma estática en el sistema. También serán utilizados los diagramas de secuencia para el análisis de los procesos, estos diagramas a su vez están incluidos dentro los diagramas de comportamiento, los cuales mostraran los procesos del sistema desde un punto de vista dinámico.

A continuación se describen los procesos actuales y semi automatizados, que son llevados a cabo dentro la Unidad Educativa Bruno Racua en las siguientes figuras.

3.2. DESCRIPCIÓN DE LA SITUACIÓN ACTUAL

En este punto se describen las etapas y flujos de información que se tienen, considerando la relación que existen entre los procesos y los usuarios, a continuación se detallan los mismos.

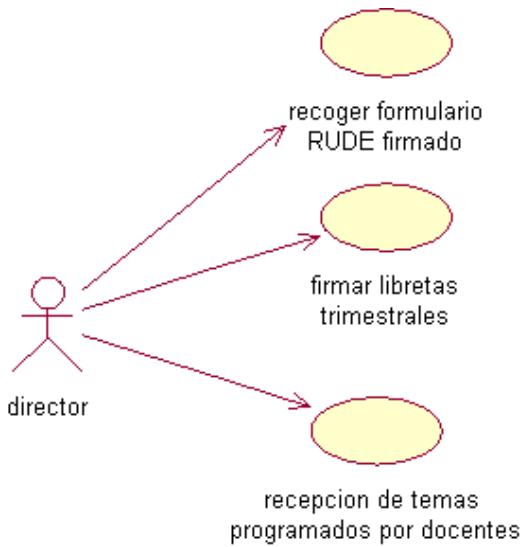
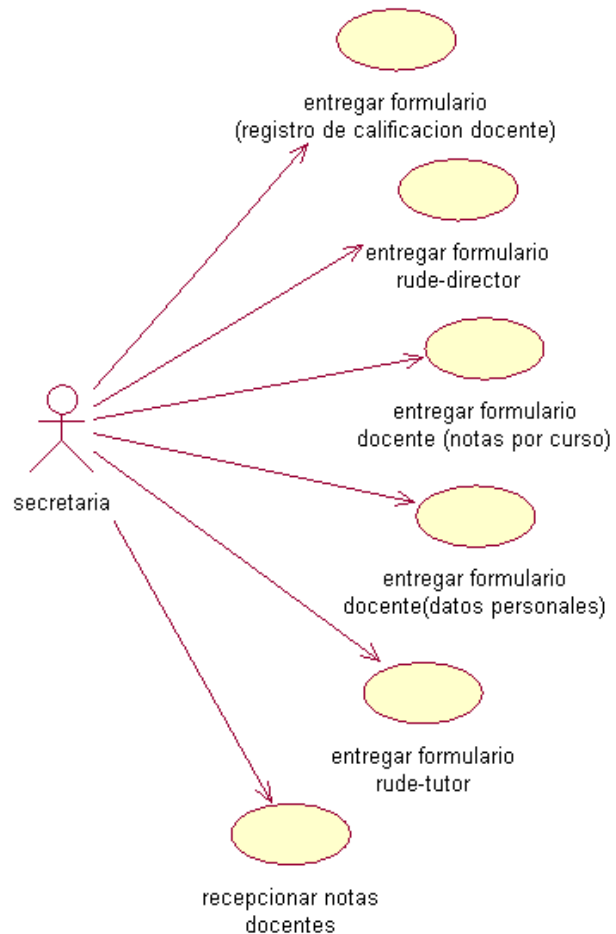


Fig. 3.1.: Proceso de registro pedagógico e inscripción de alumnos
Fuente: Elaboración propia

Fig. 3.2.: Proceso entrega de formularios
Fuente: Elaboración propia



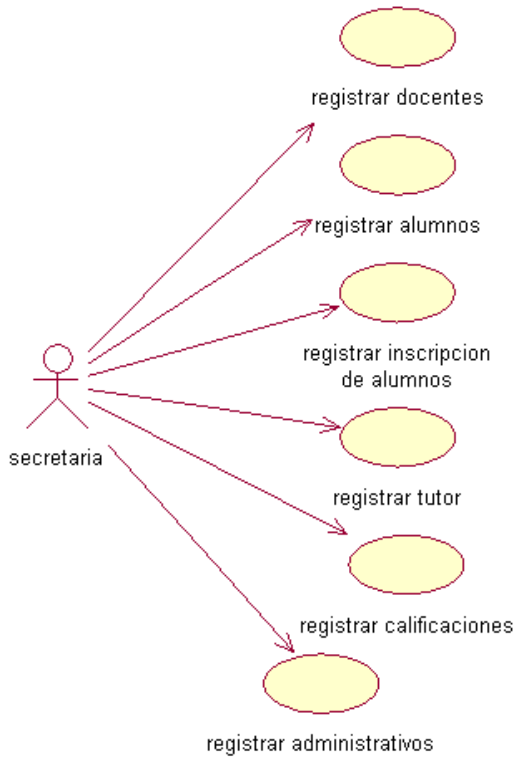
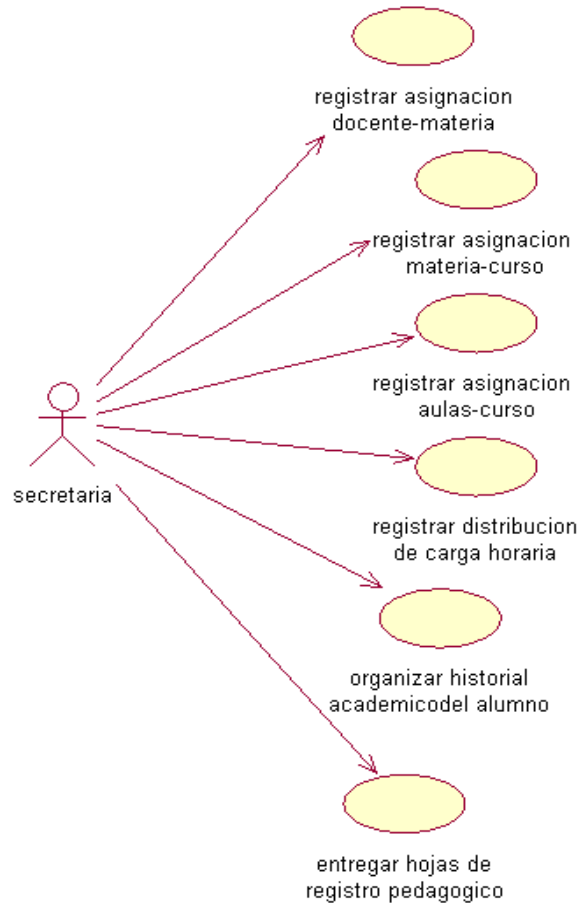


Fig. 3.3.: Proceso registro de personas y calificaciones
Fuente: Elaboración propia

Fig. 3.4.: Proceso registro asignaciones, pedagógico e historial académico
Fuente: Elaboración propia



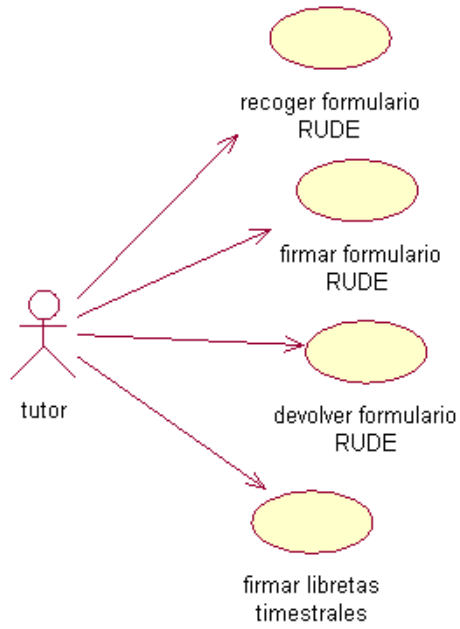
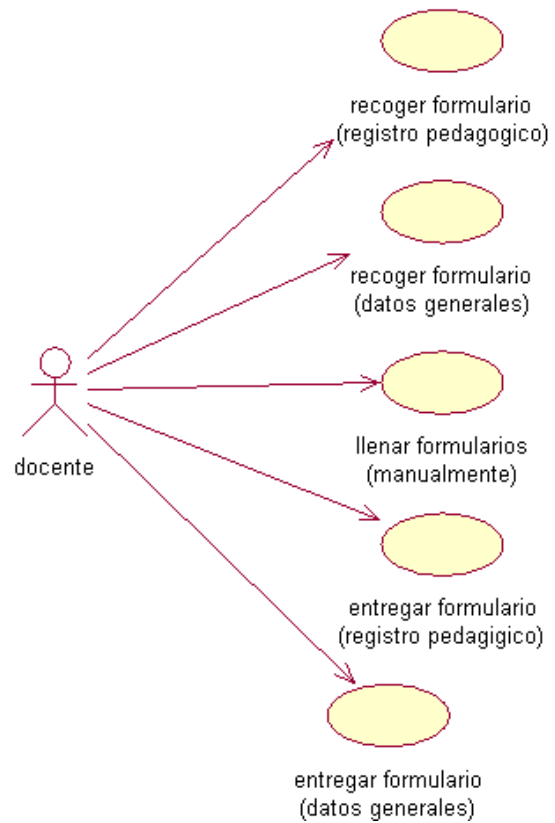


Fig. 3.6.: Proceso de registro de formularios
Fuente: Elaboración propia

Fig. 3.5.: Proceso de rubricación
Fuente: Elaboración propia



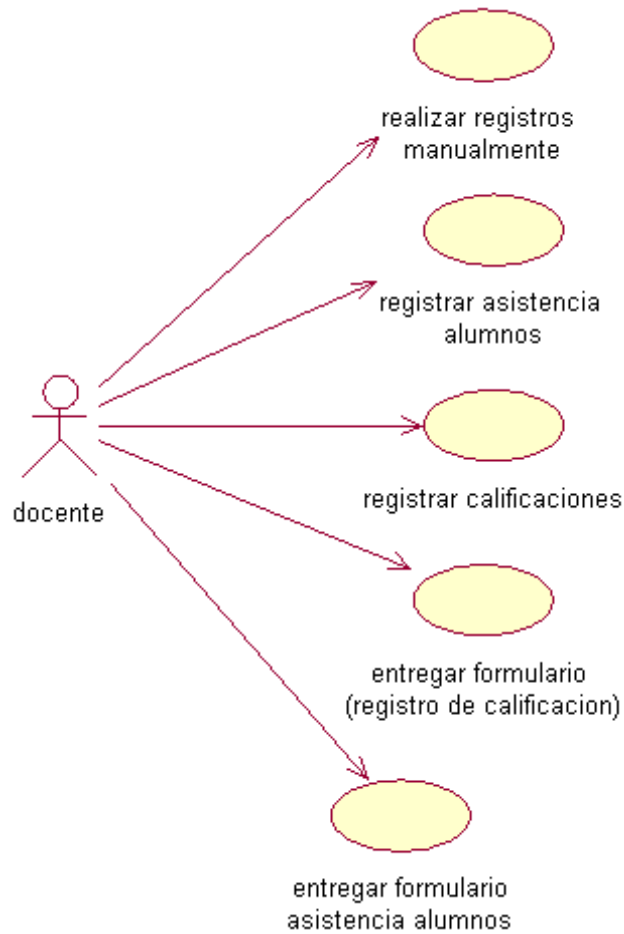


Fig. 3.7.: Procesos de registros pedagógicos
Fuente: Elaboración propia

Los procesos de: inscripción, registro de calificaciones y llenado de libreta, son procesos que se llenan en un formulario (manualmente), y posteriormente son llevados a otro formulario, esta información requiere ser trasladada con bastante cuidado para no producir fallas en los mismos y evitar demora en la organización de la información, dificultando la entrega de estos formularios con los nuevos datos en el menor tiempo posible.

3.3. APLICACION DE LA INGENIERIA INVERSA

De acuerdo al previo análisis del sistema actual y el análisis del sistema existente se aplica la ing. Inversa con los siguientes casos de uso.

3.3.1. DESCRIPCIÓN DE CASOS DE USO.

Un actor es un usuario y representa el rol genérico del sistema. El nombre que se le de a un actor deberá reflejar el papel que tendrá para el sistema. Los casos de uso son una descripción de un conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable de interés para un actor particular. Reflejan el uso que harán los actores del sistema; se muestran a través de ellos tanto las funcionalidades que ofrecerá el sistema, como los diferentes inherentes a las situaciones contempladas para cada una de estas.

Pasaremos a dar una descripción de los actores principales y sus casos de uso, dando una breve explicación de la función que realizaran dentro del sistema.

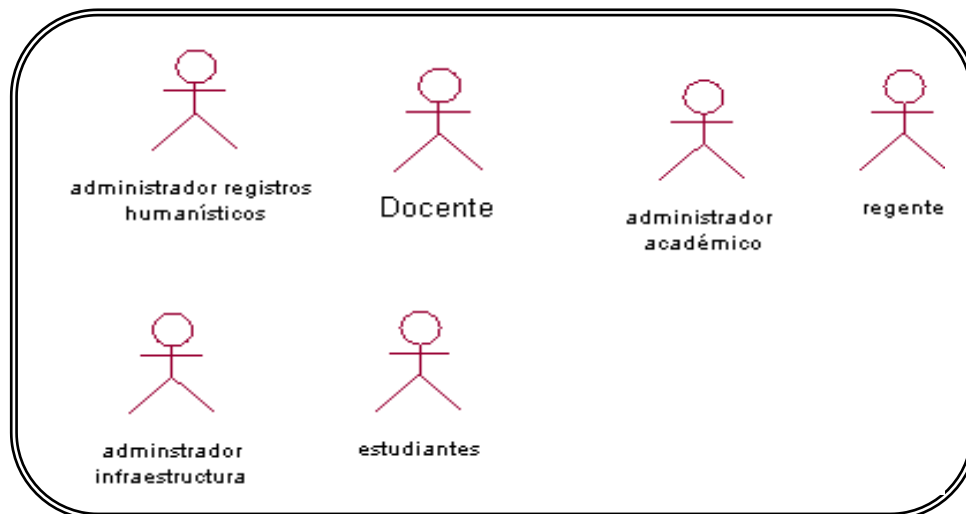


Fig. 3.3.1: Identificación de actores
Fuente: Elaboración propia

Regente.- Es el actor encargado de registrar las asistencias de los estudiantes (tipo de asistencias, fases de asistencia).

Administrador Académico.- Este actor registrara la información necesaria para el almacenamiento de datos a la base de datos de los estudiantes, realizará la asignación de docentes a los cursos, registrara horarios, datos de los docentes y plan de estudios (materias, grados, niveles).

Administrador de registro humanístico.- Es el actor encargado de la inscripción, registro de estudiantes, listado de estudiantes por paralelo, lista de horarios, impresión de libretas, listar boletín de calificaciones, notas por paralelo y autenticar la información brindada por los estudiantes.

Administrador de Infraestructura.- Es el actor que se encarga de administrar las aulas que existen dentro la unidad educativa.

Estudiante.- Este actor sólo tiene acceso al historial académico sin tener opciones de guardar, modificar o eliminar.

Docente.- El rol de este actor es el encargado de introducir las notas de los estudiantes que tenga a cargo en función a las materias que imparte, para la posterior impresión de libretas.

3.4. MODELADO EL CONTEXTO DEL SISTEMA

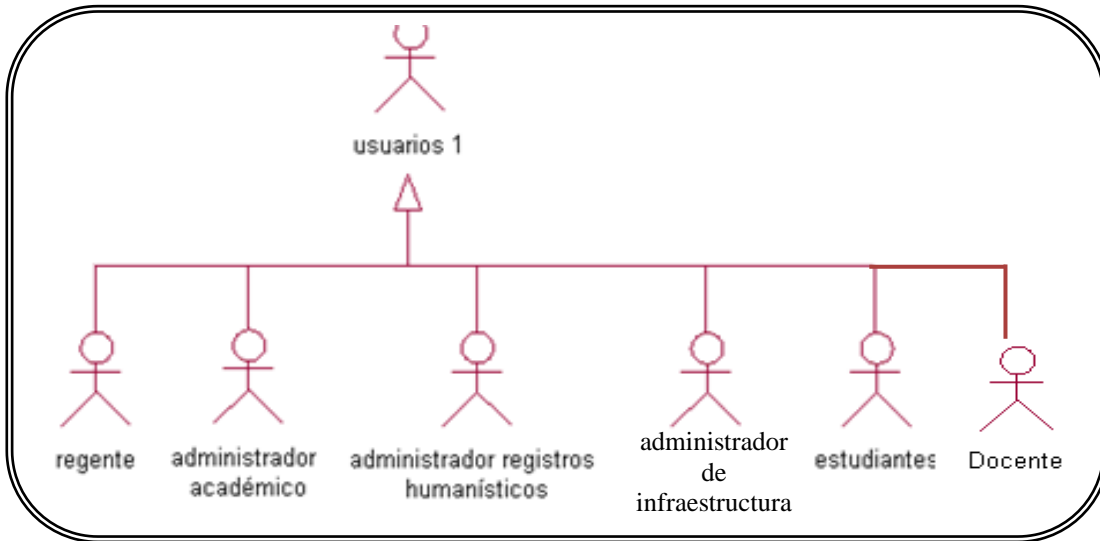


Fig. 3.3.1: Confirmación de datos de los usuarios

Fuente: Elaboración propia

a) Validar Usuarios

Caso de uso:	Validar Usuarios
Actores:	Regente, Administrador Académico, Administrador de Registros Humanísticos, Administrador de Infraestructura, Estudiante.
Resumen:	Este caso de uso es iniciado por el actores, validar usuario mediante un login y password a ser validado con su respectivo registro de usuario para así poder ingresar al sistema.
Precondiciones:	El usuario debe encontrarse previamente <i>registrado</i> como persona y pertenecer a una de las categorías de usuario.
Flujo Principal:	Se presentan usuario la pantalla principal. El usuario debe introducir nombre de usuario y contraseña (PIN) luego debe seleccionar la opción

	iniciar sesión, luego se valida el registro de usuario mediante su login y su password insertados por el usuario a la pantalla principal.
Sub Flujos:	Ninguno

Tabla 3.3.1: Caso de Uso validar usuarios

Fuente: Elaboración propia

b) Asistencia

Caso de uso:	Administrar Asistencia
Actores:	Regente
Resumen:	Registra y muestra información de asistencia de los estudiantes.
Precondiciones:	Requiere tener información sobre tipo de asistencia, fechas de asistencia, grado, paralelo.
Flujo Principal:	Se presenta al usuario la pantalla asistencia de estudiantes Esta pantalla lista a los estudiantes por curso, dando la opción de su tipo de asistencia y si tiene algún retraso.
Sub Flujos:	Grado Se debe tener definida la gestión, período e introducir la clave correspondiente, luego se debe presionar la opción buscar. La cual nos envía a registrar grado paralelos. Registra grado, paralelos. Se debe tener información de los estudiantes por grados, y los paralelos de cada curso.

Tabla 3.3.2: Caso de Uso administrar asistencias

Fuente: Elaboración propia

c) Personal

Caso de uso:	Administrar personas-cargos
Actores:	Administrador humanístico
Resumen:	Registra y muestra información de los tipos de ítems y cargo del personal administrativo.
Precondiciones:	Requiere tener información sobre personas, cargos, tipo de Items, la institución y tipo de financiamiento.
Flujo Principal:	Presenta al usuario la pantalla administrar personas- cargos. Esta pantalla contiene información del personal administrativo. Código de Items, puntaje y fechas de ingreso
Sub Flujos:	Administrar tipo y Items Administrar cargos

Tabla 3.3.3: Caso de Uso Administrar personas-cargos

Fuente: Elaboración propia

d) Bonos

Caso de uso:	Administrar Bonos
Actores:	Administrador de registros humanísticos
Resumen:	Registra y muestra información de los diferentes bonos que perciben los estudiantes.
Precondiciones:	Requiere tener información sobre, tipo de bonos, grado, paralelo.
Flujo Principal:	Se presenta al usuario la pantalla bonos estudiantes. Esta pantalla contiene información del estudiante por curso, grado y paralelo su tipo de bono y la fecha de entrega del bono.
Sub Flujos:	Grado Se debe tener definida la gestión, período e introducir la clave

	<p>correspondiente, luego se debe presionar la opción buscar. La cual nos envía a seleccionar los datos.</p> <p>Selecciona los datos.</p> <p>Se debe tener información de los estudiantes por grados, y los paralelos de cada curso y el tipo de bono que percibe.</p>
--	--

Tabla 3.3.4: Caso de Uso Administrar bonos
Fuente: Elaboración propia

e) Encuesta Social

Caso de uso:	Encuesta social
Actores:	Administrador de registros humanísticos
Resumen:	Registra y muestra información del entorno social de los estudiantes.
Precondiciones:	Requiere tener información sobre período (ingresa los datos), registro de estudiantes para su búsqueda (CI, nombres).
Flujo Principal:	Se presenta al usuario la pantalla listar estudiantes y brinda la opción del registro socioeconómico del estudiante, en esta se registra información del estudiante (idioma, servicios básicos, salud, empleo, medios de comunicación y transporte).
Sub Flujos:	<p>Grado</p> <p>Se debe tener definida la gestión, período e introducir la clave correspondiente, luego se debe presionar la opción buscar. La cual nos envía a búsqueda de estudiantes.</p> <p>Búsqueda de estudiantes.</p> <p>Se debe tener información de los estudiantes.</p>

Tabla 3.3.5: Caso de Uso Encuesta social
Fuente: Elaboración propia

f) Apoderados

Caso de uso:	Registrar apoderados
Actores:	Administrador de registros humanísticos
Resumen:	Registra y muestra información de los diferentes bonos que perciben los estudiantes.
Precondiciones:	Requiere tener información sobre período (ingresa los datos), registro del estudiante para la búsqueda (CI, nombres).
Flujo Principal:	Se presenta al usuario una lista de estudiante. Esta pantalla brinda la opción del registro del tutor o padre de familia al momento de la inscripción: almacena información tutor o padre de familia (parentesco, CI, apellidos, nombres, ocupación laboral, mayor grado de instrucción, origen étnico, idiomas que habla frecuentemente).
Sub Flujos:	Grado Se debe tener definida la gestión, período e introducir la clave correspondiente, luego se debe presionar la opción buscar. La cual nos envía a búsqueda de estudiantes. Búsqueda de estudiantes Se debe tener información de los estudiantes.

Tabla 3.3.6: Caso de Uso Registrar apoderados

Fuente: Elaboración propia

g) Estudiantes

Caso de uso:	Administrar Estudiantes por paralelo
Actores:	Administrador de registros humanísticos
Resumen:	Registra y muestra información de los estudiantes

Precondiciones:	Requiere tener información sobre tipo de estudiantes, de estar registrada como persona, la carrera, el tipo de admisión, el tipo de estudiante.
Flujo Principal:	Se presenta al usuario la pantalla administrar estudiantes .Esta pantalla brinda y almacena información del estudiante. Fecha de inscripción, fecha de egreso y el id_rol
Sub Flujos:	ninguno

Tabla 3.3.7: Caso de Uso Administrar Estudiantes por paralelo
Fuente: Elaboración propia

h) Horarios

Caso de uso:	Listar horarios
Actores:	Administrador de registros humanísticos
Resumen:	Muestra información de los horarios asignados a los cursos
Precondiciones:	Requiere tener información sobre carrera, turnos y clave
Flujo Principal:	Se presenta al usuario la pantalla ver horarios Esta pantalla brinda información todos los horarios de los cursos.
Sub Flujos:	ninguno

Tabla 3.3.8: Caso de Uso Listar horarios
Fuente: Elaboración propia

i) Inscripciones

Caso de uso:	Inscribir alumno nuevo
Actores:	Administrador de registros humanísticos, administrador de registros técnicos
Resumen:	Realiza la inscripción de estudiantes nuevos.

Precondiciones:	Requiere tener información sobre período, gestión, registro de estudiante, ubicaciones (país, dpto., provincia, sección, municipio, cantón, localidad, zona, estado civil, sexo, grupo sanguíneo, carrera, tipo admisión, nivel, grado, paralelo).
Flujo Principal:	Se presenta al usuario la pantalla inscripción de alumno nuevo. En esta pantalla se registrarán los datos personales (nombre, apellidos, ci, etc.), y los documentos que presenta en el momento de la inscripción.
Sub Flujos:	ninguno

Tabla 3.3.9: Caso de Uso Inscribir alumno nuevo
Fuente: Elaboración propia

Caso de uso:	Inscribir alumno por traspaso
Actores:	Administrador de registros humanísticos, administrador de registros técnicos
Resumen:	Realiza la inscripción de estudiantes.
Precondiciones:	Requiere tener información sobre período, gestión, registro de estudiante, carrera.
Flujo Principal:	Se presenta al usuario la pantalla inscripción por traspaso. En esta pantalla se registrarán los datos personales carrera y tipo de admisión.
Sub Flujos:	Ninguno

Tabla 3.3.10: Caso de Uso Inscribir alumno por traspaso
Fuente: Elaboración propia

Caso de uso:	Inscribir alumno antiguo
Actores:	Administrador de registros humanísticos, administrador de registros técnicos
Resumen:	Realiza la inscripción de estudiantes antiguos
Precondiciones:	Requiere tener información sobre período, gestión, registro de estudiante, carrera, nivel, grado, paralelo.
Flujo Principal:	Se presenta al usuario la pantalla datos por admisión. En esta pantalla se registrarán los datos personales carrera y tipo de admisión.
Sub Flujos:	Ninguno

Tabla 3.3.11: Caso de Uso Inscribir alumno antiguo

Fuente: Elaboración propia

j) Libretas

Caso de Uso:	Administrar Libretas
Actores:	Administrador de registros humanísticos
Resumen:	Realiza el cargado de notas en las libretas.
Precondiciones:	Requiere tener información sobre período, gestión, inscripción del estudiante (carrera).
Flujo Principal:	Se presenta al usuario la pantalla centralizador de notas por alumno. En esta pantalla se seleccionan los datos del estudiante (carrera, nivel, grado y paralelo) muestra una lista de todos los estudiantes, luego se seleccionan a los estudiantes que se desea realizar el reporte se presiona siguiente y demuestran reportes del Su libreta.
Sub Flujos:	Abrir fase Deben estar previamente registradas carrera, nivel, grado.

Observaciones:	Si la información es incompleta en el registro, se vuelve a solicitar al usuario para que complete los datos obligatorios, así realizara la búsqueda, posteriormente desplegara los diferentes centralizadores de notas que existen.
-----------------------	--

Tabla 3.3.12: Caso de Uso administrar libretas
Fuente: Elaboración propia

k) Parámetros Sociales

Caso de uso:	Parámetros sociales
Actores:	Administrador de registros humanísticos
Resumen:	Este caso de uso es iniciado por el usuario sistema ya validado, realiza la lista y nuevo registros de los parámetros sociales.
Precondiciones:	Ninguna
Flujo Principal:	Se presenta al usuario la pantalla parámetros sociales. En esta pantalla se seleccionan los dos nuevos parámetros que se desea registrar (tipo lenguas, tipo de entidades, tipo medios de comunicación, tipo actividades laborales, tipo centros de salud, tipo transportes, tipo discapacidades, tipo servicios básicos, tipo origen discapacidades, tipo relaciones parentesco, tipo grados instrucción y tipo origen étnico) muestra una lista de todos los datos existentes.
Sub Flujos:	Ninguno

Tabla 3.3.13: Caso de Uso Parámetros sociales
Fuente: Elaboración propia

l) Vacunas

Caso de uso:	Vacunas
Actores:	Administrador de registros humanísticos
Resumen:	Este caso de uso es iniciado por el usuario sistema ya validado, realiza la lista de vacunas que tienen los estudiantes.
Precondiciones:	Administrar tipo vacunas
Flujo Principal:	Se presenta al usuario la pantalla vacunas estudiantes. En esta pantalla se seleccionan el tipo de vacunas, grado, paralelo, se seleccionan siguiente para que muestre una lista de las vacunas por estudiante
Sub Flujos:	Administrar tipo vacunas se registra el tipo vacunas y su descripción

Tabla 3.3.14: Caso de Uso Vacunas

Fuente: Elaboración propia

m) Asignaciones

Caso de uso:	Asignaciones docentes
Actores:	Administrador académico
Resumen:	Este caso de uso es iniciado por el usuario sistema ya validado, realiza la selección de datos, buscan docentes y luego le asigna un curso.
Precondiciones:	La persona además también debe estar registrado como docentes
Flujo Principal:	Se presenta al usuario la pantalla asignaciones docentes, luego se realiza la búsqueda del docente. En esta pantalla se seleccionan el paralelo, grado, tipo de transacción, tipo docente.
Sub Flujos:	Registro del docente se registra a la persona como docentes

Tabla 3.3.15: Caso de Uso Asignaciones docentes

Fuente: Elaboración propia

n) Docentes

Casos de uso:	Administrar docentes
Actores:	Administrador académico
Resumen:	Este caso de uso es iniciado por el usuario sistema ya validado, realiza la selección de la persona y luego la asigna como docentes.
Precondiciones:	Docentes debe estar asignado en tipo formaciones y tipo docentes
Flujo Principal:	Se presenta al usuario la pantalla asignaciones docentes, luego se realiza la búsqueda del docente. En esta pantalla se seleccionan el paralelo, grado, tipo de transacción, tipo docente.
Sub Flujos:	Registro del docente se registra a la persona como docentes

Tabla 3.3.16: Caso de Uso Administrar docentes
Fuente: Elaboración propia

3.5. DIAGRAMA DE CASOS DE USO DEL SISTEMA

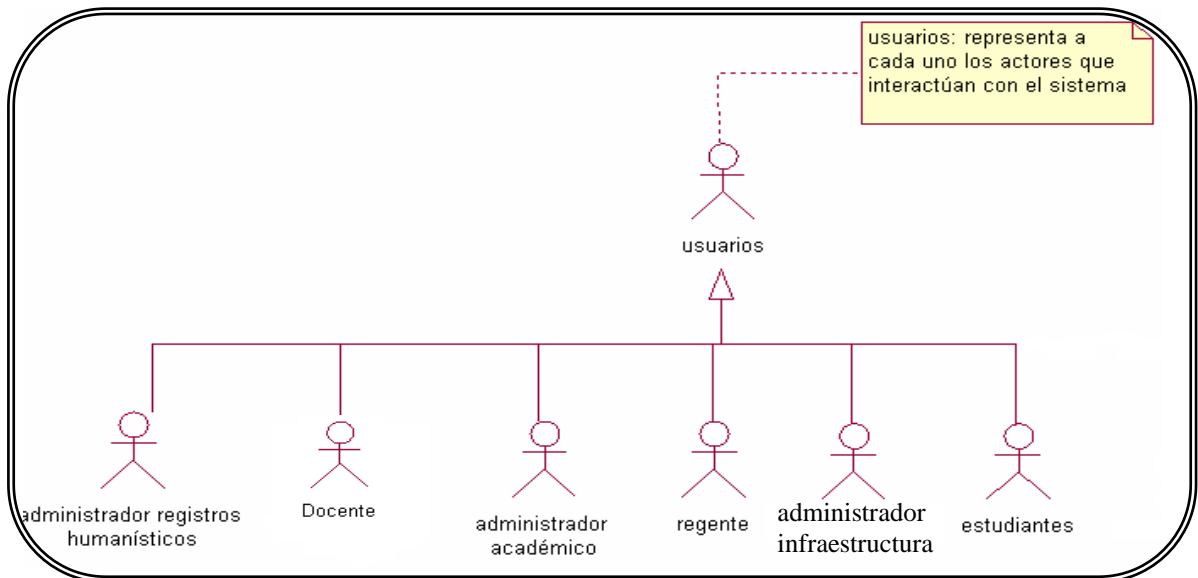


Figura 3.5: Caso de uso relación entre los actores y el usuario del sistema
Fuente: Elaboración propia

VISTA GLOBAL DE LOS CASOS DE USO Y LOS ACTORES

3.5.1. Vista del rol del regente

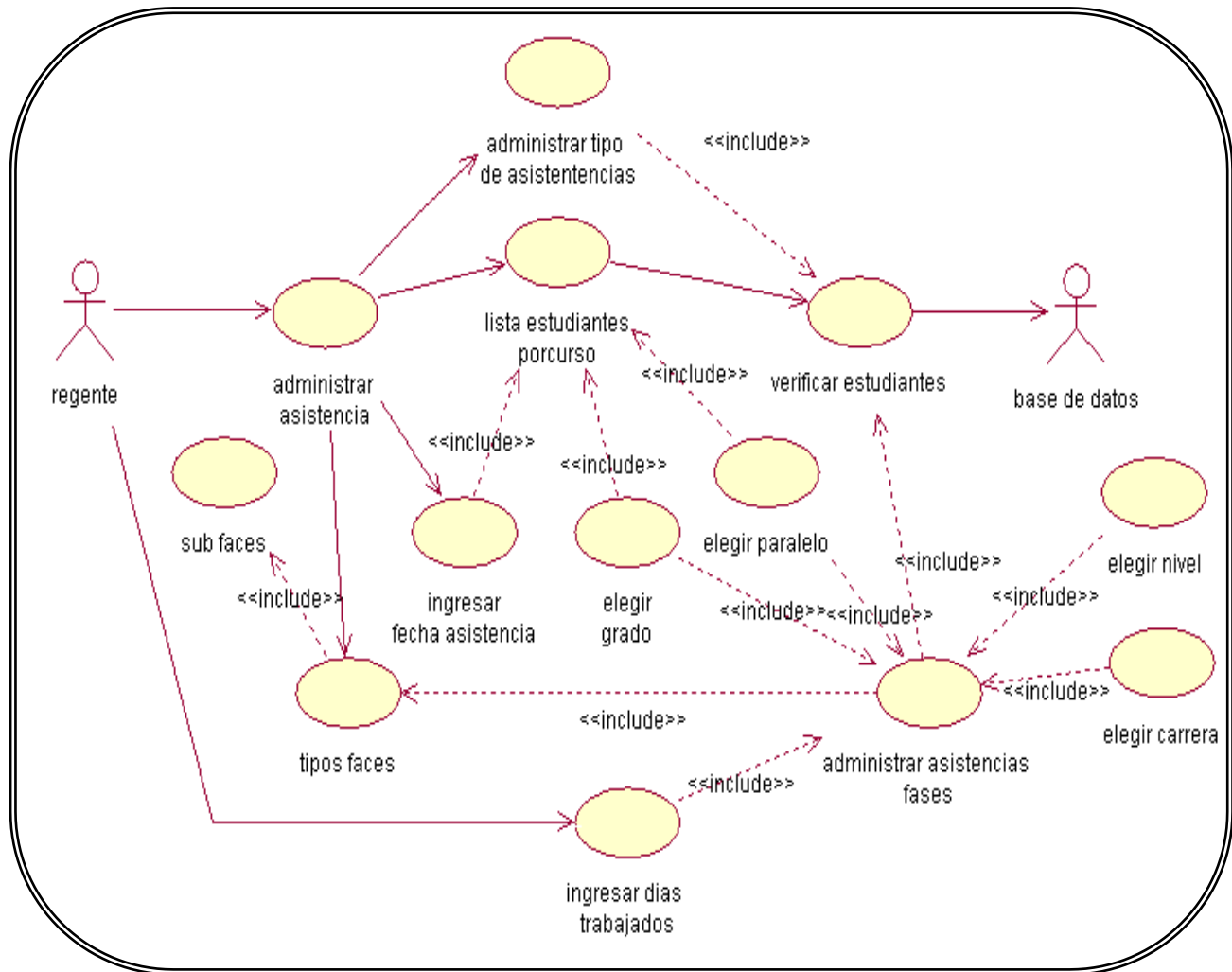


Figura. 3.5.1: Vista del rol del regente

Fuente: Elaboración propia

3.5.2. Vista del rol del Administrador Académico

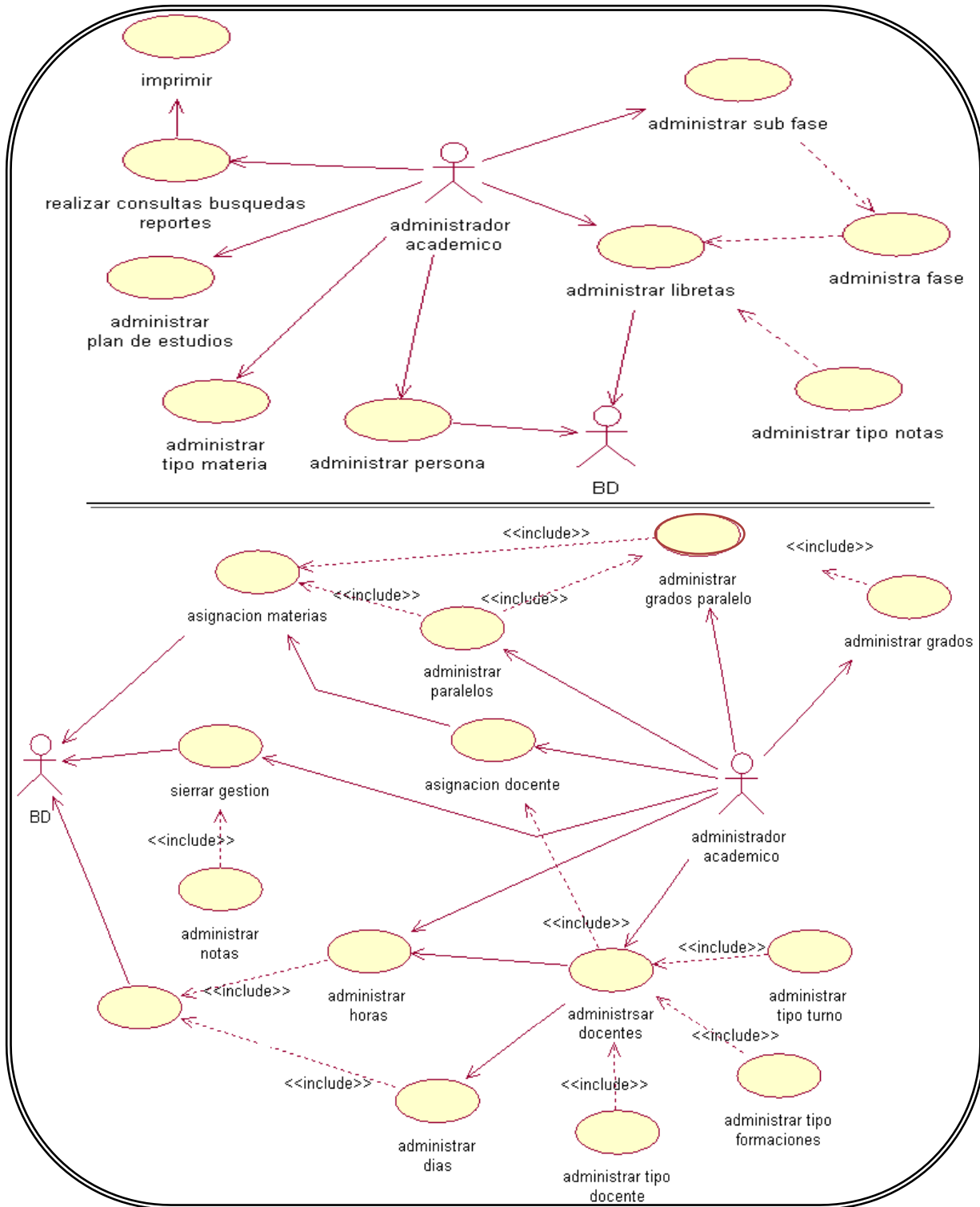


Figura. 3.5.2: Vista del rol del administrador de registro académico
Fuente: Elaboración propia

3.5.3. Vista del rol del Administrador de Registro Humanístico

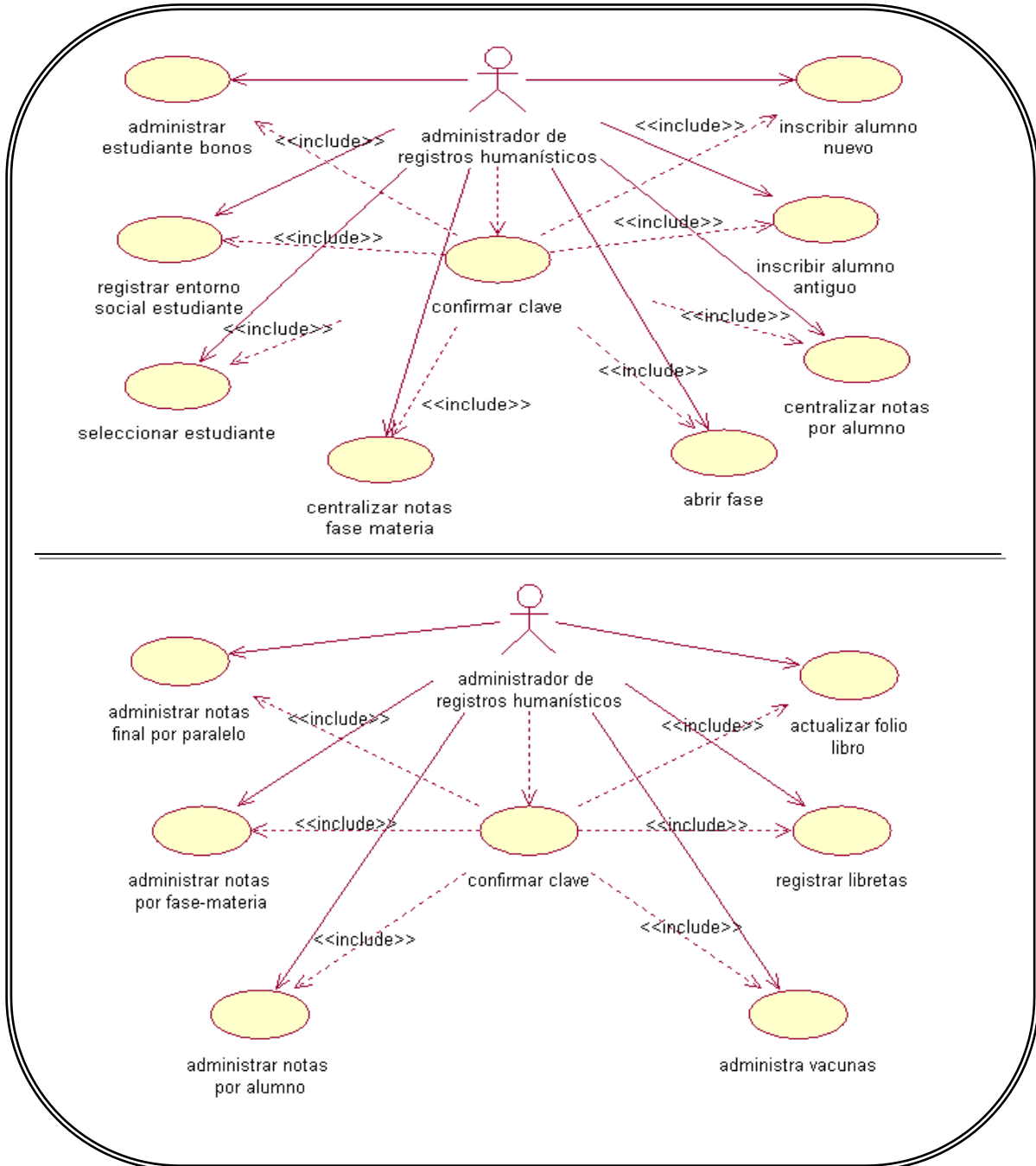


Figura. 3.5.3: Vista del rol del administrador humanístico N°1

Fuente: Elaboración propia

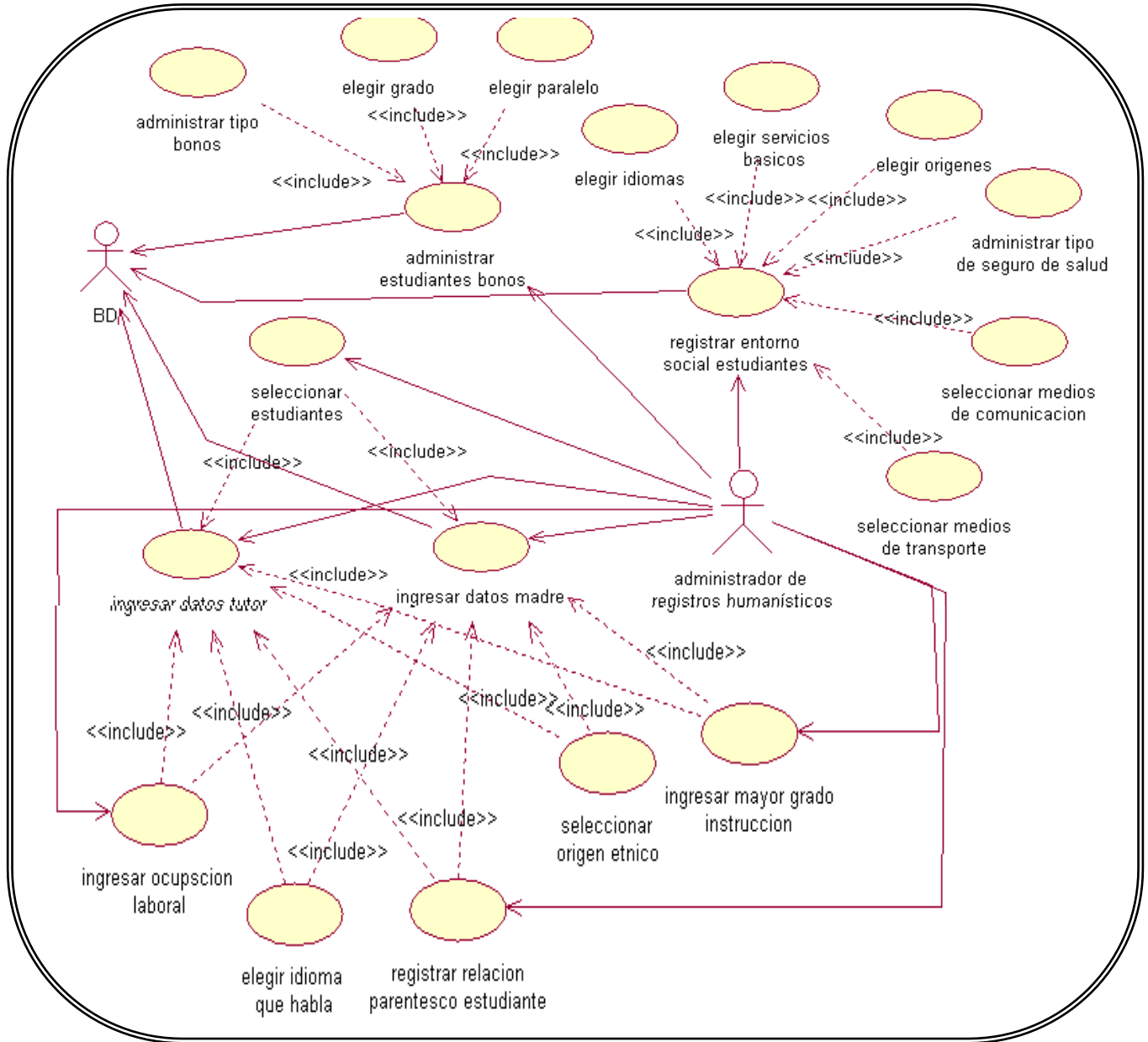


Figura. 3.5.3.1: Vista del rol del administrador humanístico N°2
Fuente: Elaboración propia

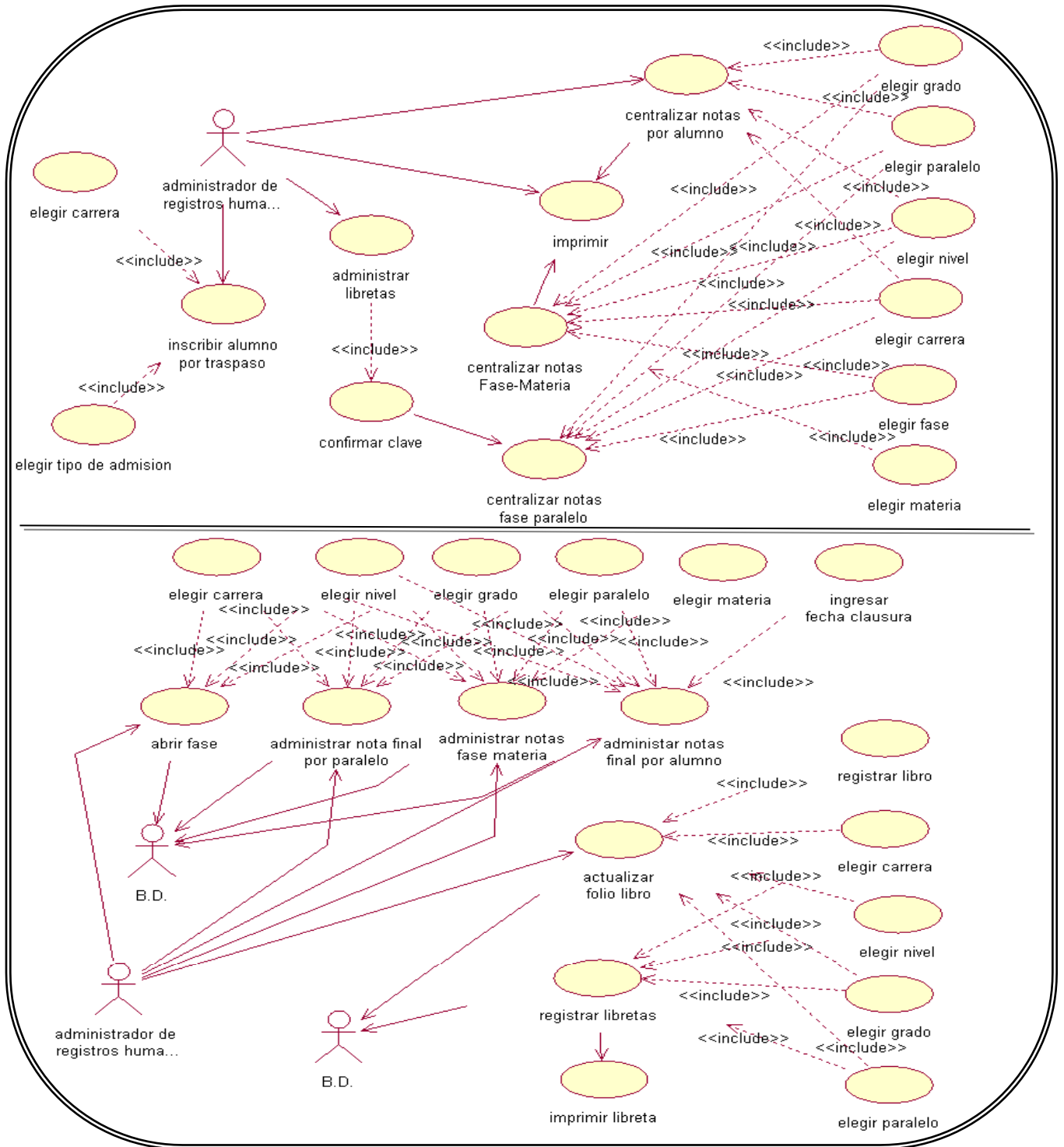


Figura. 3.5.3.2: Vista del rol del administrador humanístico N°4

Fuente: Elaboración propia

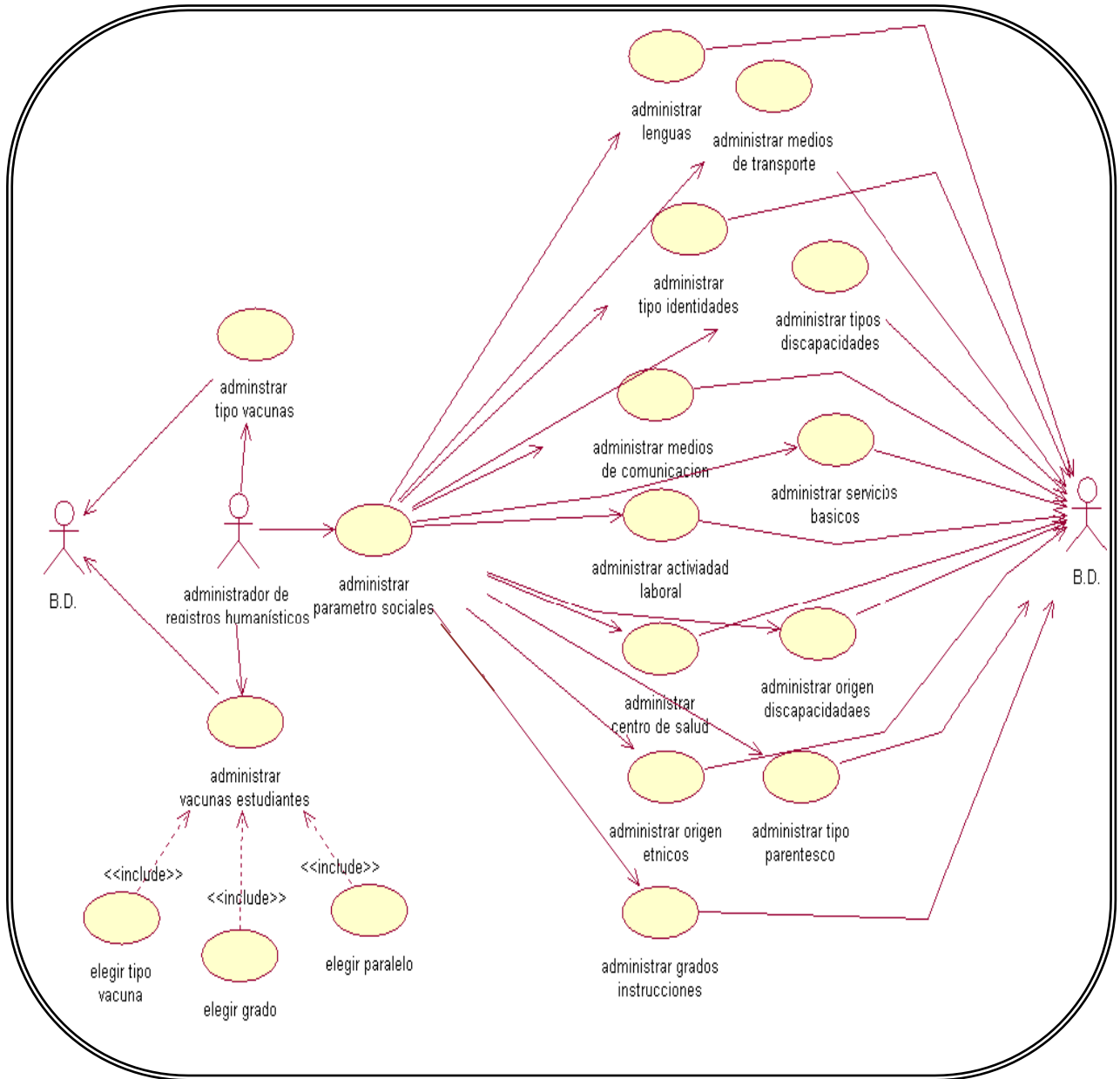


Figura. 3.5.3.3: Vista del rol del administrador humanístico N°5

Fuente: Elaboración propia

3.5.4. Vista del rol del Administrador de Infraestructura

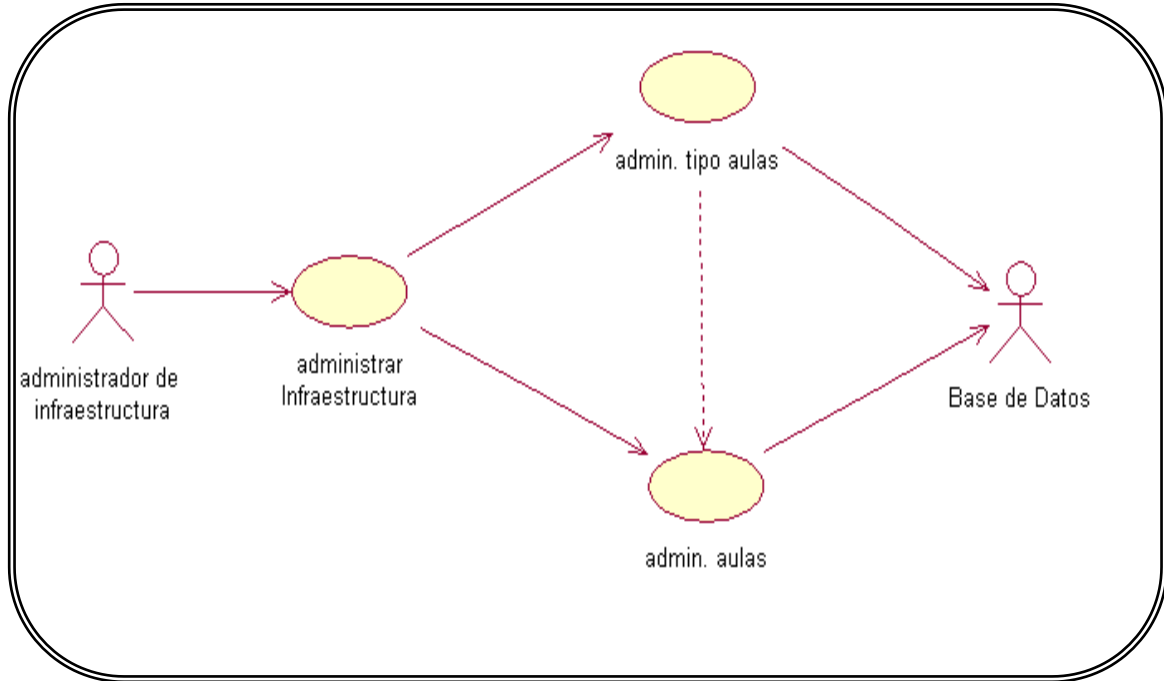


Figura. 3.5.4: Vista del rol del administrador de infraestructura
Fuente: Elaboración propia

3.5.5. Vista del rol del Estudiante

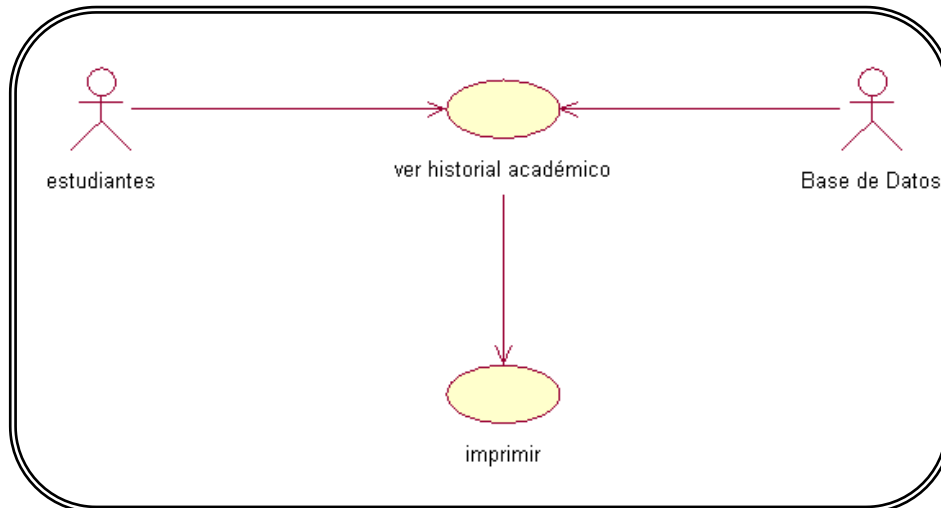


Fig. 3.5.5: Vista del rol del estudiante
Fuente. Elaboración propia

3.5.6. Vista del rol del Docente

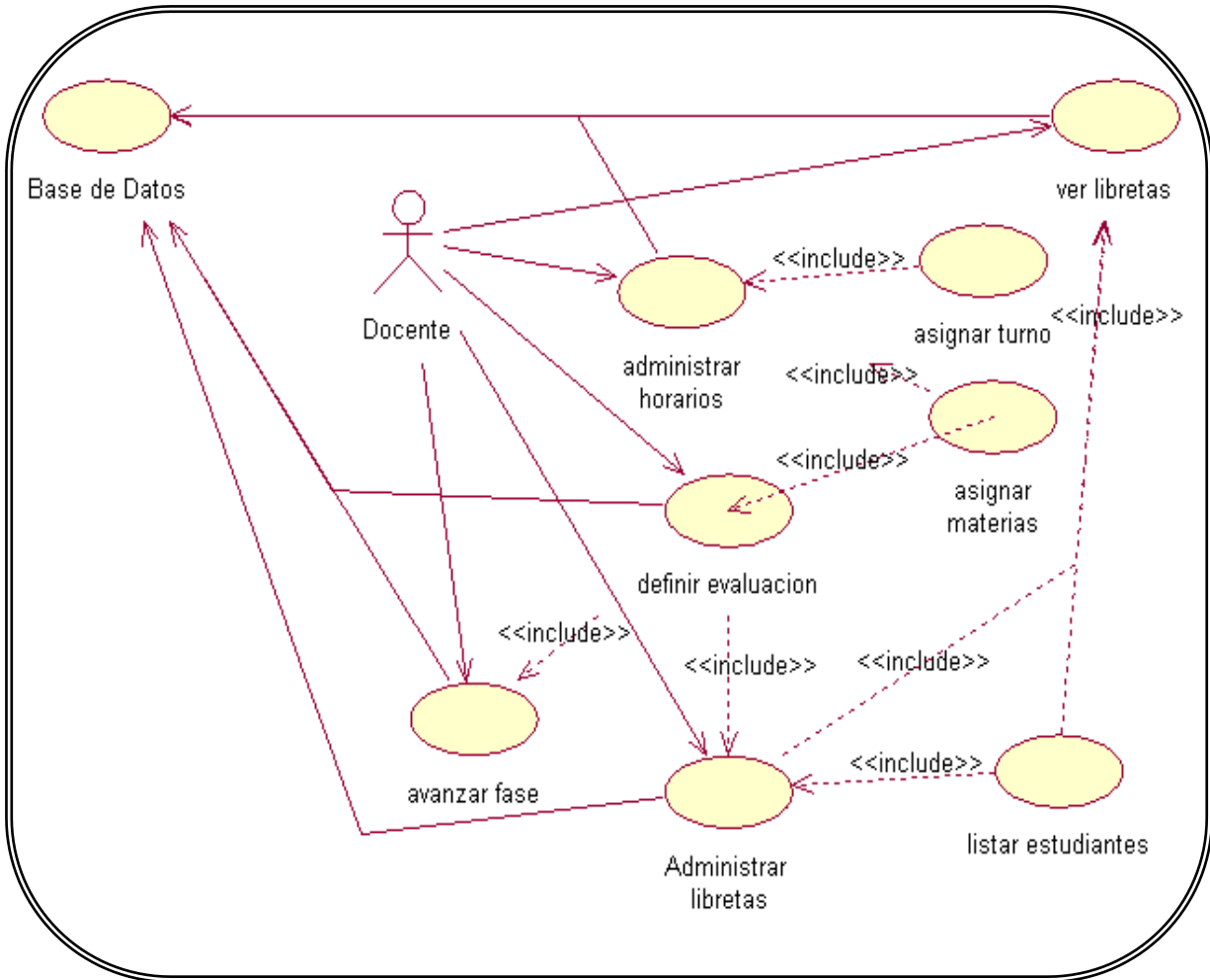


Fig. 3.5.6: Vista del rol del Docente

Fuente. Elaboración propia

3.6. DIAGRAMAS DE SECUENCIA

De acuerdo a la metodología y el objetivo planteado sobre el análisis y documentación del sistema de información, a continuación se describen los *diagramas de secuencia*, para representar los procesos dinámicos, mediante los cuales se definirá la estructura y comportamiento del sistema.

En la siguiente tabla se especifican los nombres completos de los mensajes que se envían entre los objetos.

Nº	NOMENCLATURA	SIGNIFICADO
1	LEA	Listar asistencia estudiante
2	IAE	Insertar asistencia estudiantes
3	LEAF	Listar estudiante asistencia fase
4	MAEF	Modificar asistencia estudiante fase
5	LAMD	Listar Asignación Materias Docente
6	LPG	Listar Paralelos Grados
7	IAD	Insertar Asignación Docente
8	LAMD	Listar Asignación Materias Docente
9	LCF	Listar Carreras Fases
10	CCG	Confirmar cerrar gestión
11	ICG	Insertar cerrar gestión
12	LTT	Listar tipo turnos
13	VU	Verificar usuario
14	LH	Listar Horarios
15	BDHM	Buscar Días Horas Materias
16	IH	Insertar Horarios
17	MH	Modificar Horarios
18	EH	Eliminar Horarios
19	LFG	Listar Fases Generales
20	LFSFN	Listar Fases Sub Fases Nivel
21	LF	Listar Fases
22	IF	Insertar Fases
23	ISF	Insertar Sub Fases
24	LSFF	Listar Sub Fases Fases

25	LP	Listar paralelos
26	LPa	Listar país
27	Ldpto	Listar departamentos
28	LProv	Listar Provincias
29	Lsec	Listar Secciones
30	LCa	Listar Cantones
31	Llo	Listar Localidades
32	LEC	Listar Estados Civiles
33	LTGA	Listar Tipos Grupos Sanguíneos
34	LPN	Listar Personas Nombres
35	LPP	Listar Paralelos Periodo
36	LEPB	Listar estudiante paralelo bono
37	IBE	Insertar Bono Estudiante
38	LE	Listar Estudiantes
39	LTL	Listar Tipos Lenguas
40	IPLM	Insertar Prs Lenguas Maternas
41	LTI	Listar Tipos Identidades
42	IPI	Insertar Prs Identidades
43	IPLE	Insertar Prs Lenguas Estudiantes
44	LTCC	Listar Tipos Centros Salud
45	IPCS	Insertar Prs Centros Salud
46	LTD	Listar Tipos Discapacidades
47	IPD	Insertar Prs Discapacidades
48	LTOD	Listar Tipos Origen Discapacidades
49	IPOD	Insertar Prs Origen Discapacidades
50	LTSB	Listar Tipos Servicios Básicos
51	IPSB	Insertar Prs Servicios Básicos
52	LTAL	Listar Tipos Actividades Laborales
53	IPAL	Insertar Prs Actividades Laborales
54	LTMC	Listar Tipos Medios Comunicaciones
55	IPMC	Insertar Prs Medios Comunicaciones
56	LTMT	Listar Tipos Medios Transportes
57	IPMT	Insertar Prs Medios Transportes
58	LRP	Listar Relaciones Parentesco
59	LTGI	Listar Tipos Grados Instrucciones
60	LTOE	Listar Tipos Orígenes Étnicos
61	BTGRP	Buscar Tipos Relaciones Parentesco
62	BTGI	Buscar Tipos Grados Instrucciones

63	IPAP	Insertar Prs Apoderados Padre
64	IPAM	Insertar Prs Apoderados Madre
65	IEP	Insertar Estudiante Plan
66	LAulas	Listar Aulas
67	LTAsistencias	Listar Tipos Asistencias
68	LEA	Listar Estudiantes Asistencias
69	VU	Verificar Usuarios
70	IAE	Insertar Asistencia Estudiante

Tabla 3.5.1. Nomenclatura de los mensajes que se envían entre objetos
Fuente. Elaboración propia

3.6.1. Diagrama de secuencia para acceder al sistema

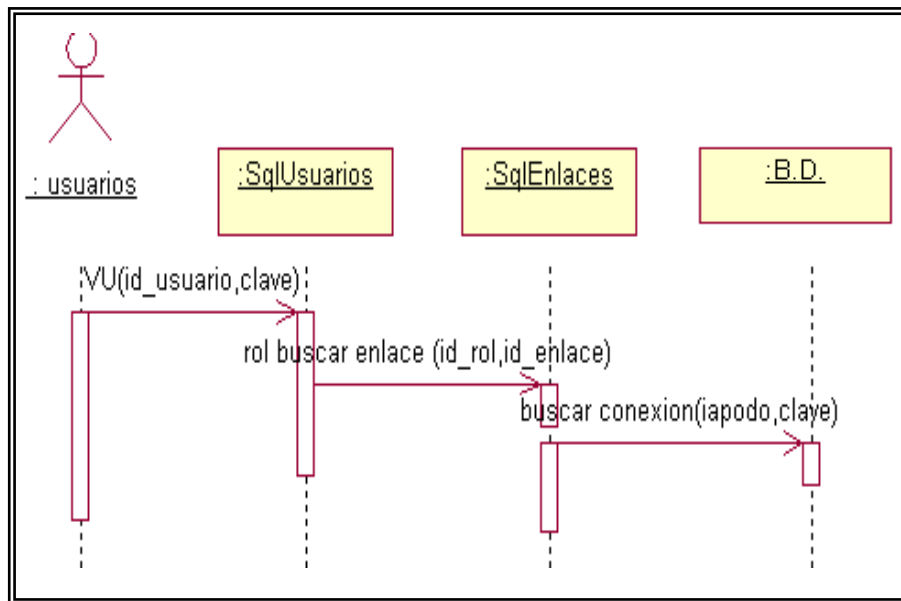


Fig. 3.6.1: Diagrama de secuencia acceso al sistema
Fuente. Elaboración propia

3.6.2. Diagrama de secuencia desde el punto de vista del Regente

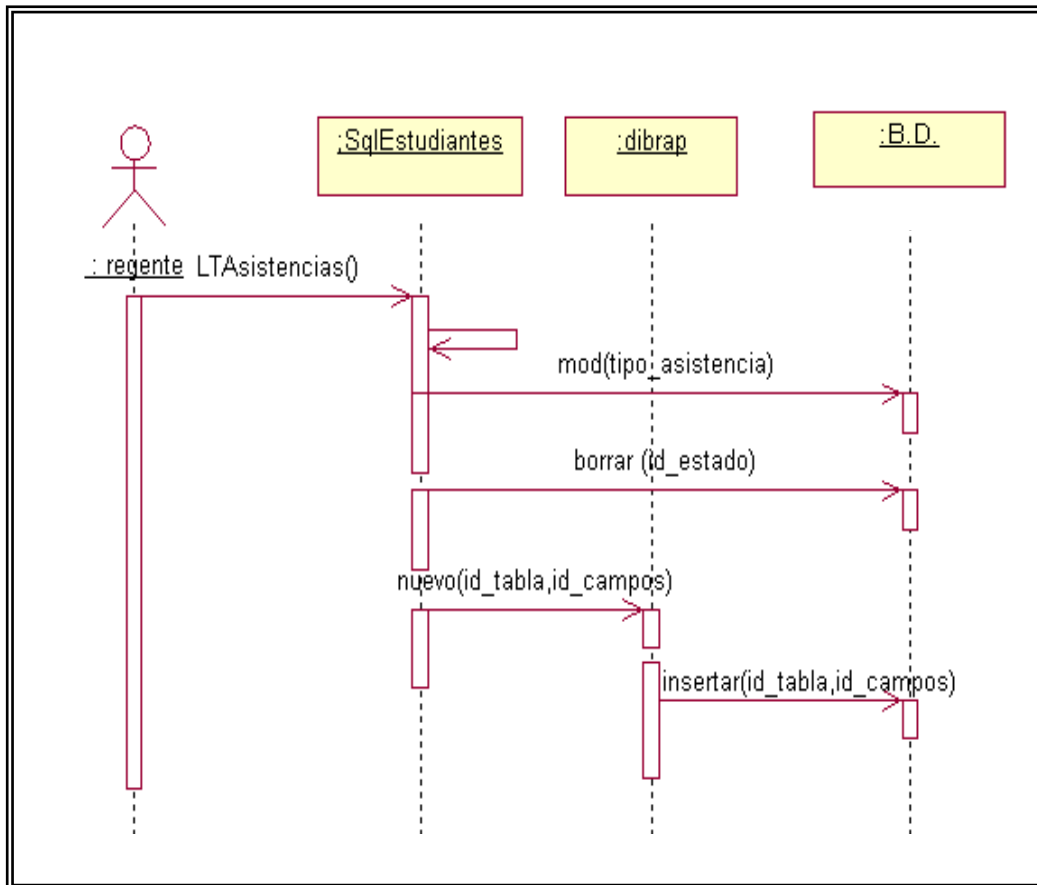


Fig. 3.6.2.1: Diagrama de secuencia Administrar tipos de asistencias

Fuente. Elaboración propia

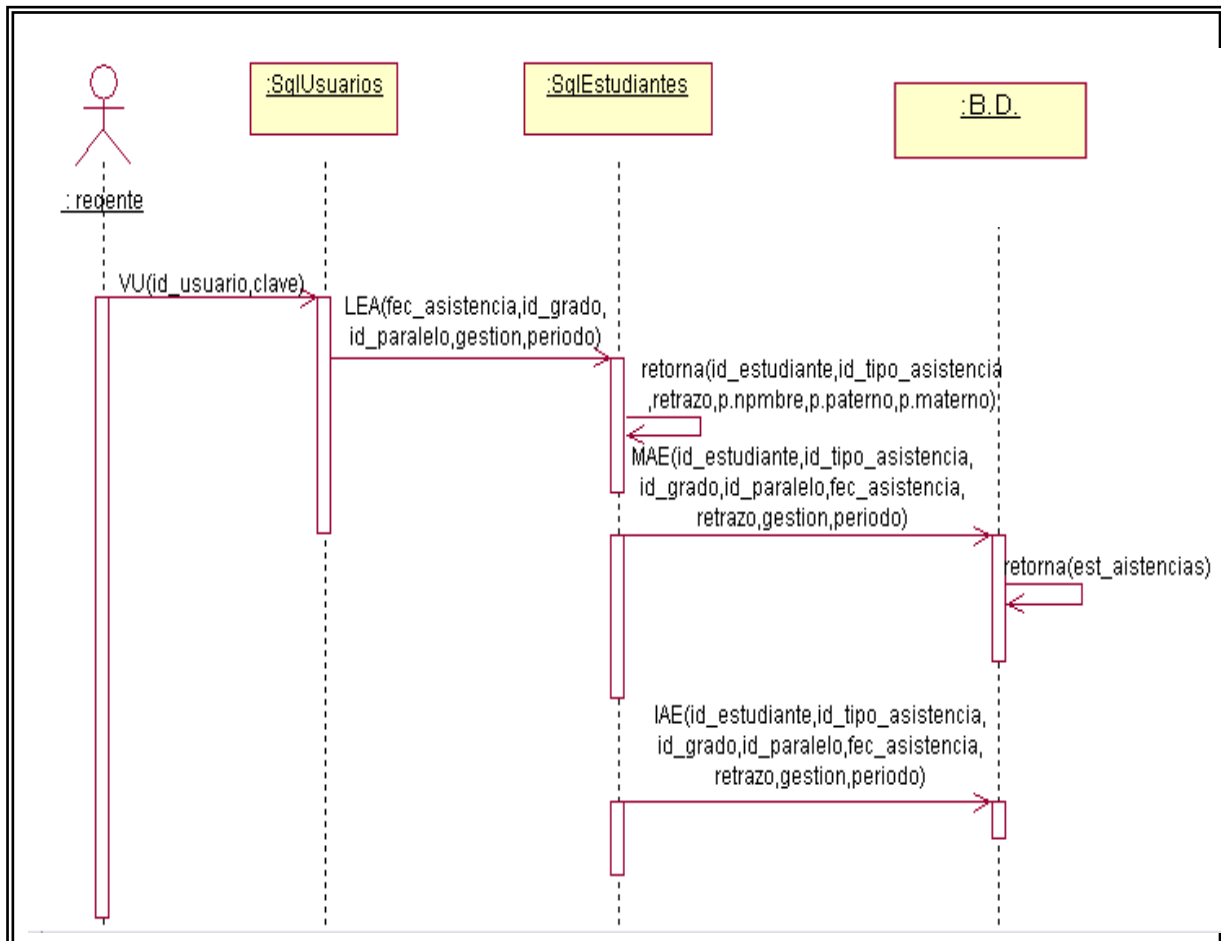


Fig. 3.6.2.2: Diagrama de secuencia Asistencia Estudiantes
Fuente. Elaboración propia

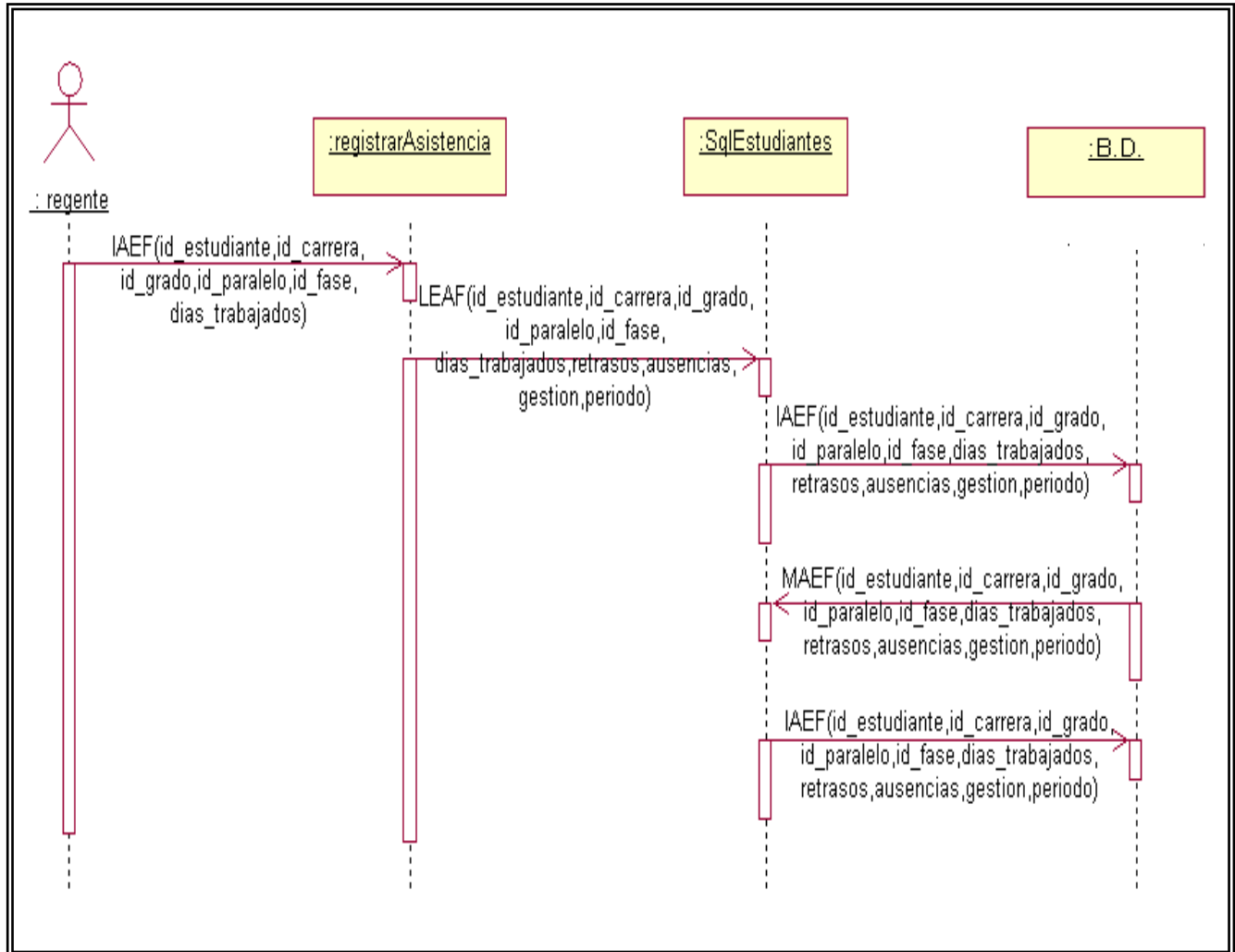


Fig. 3.6.2.3: Diagrama de secuencia Registro Datos Apoderados

Fuente. Elaboración propia

3.6.3. Diagrama de secuencia desde punto de vista del Administrador de Infraestructura

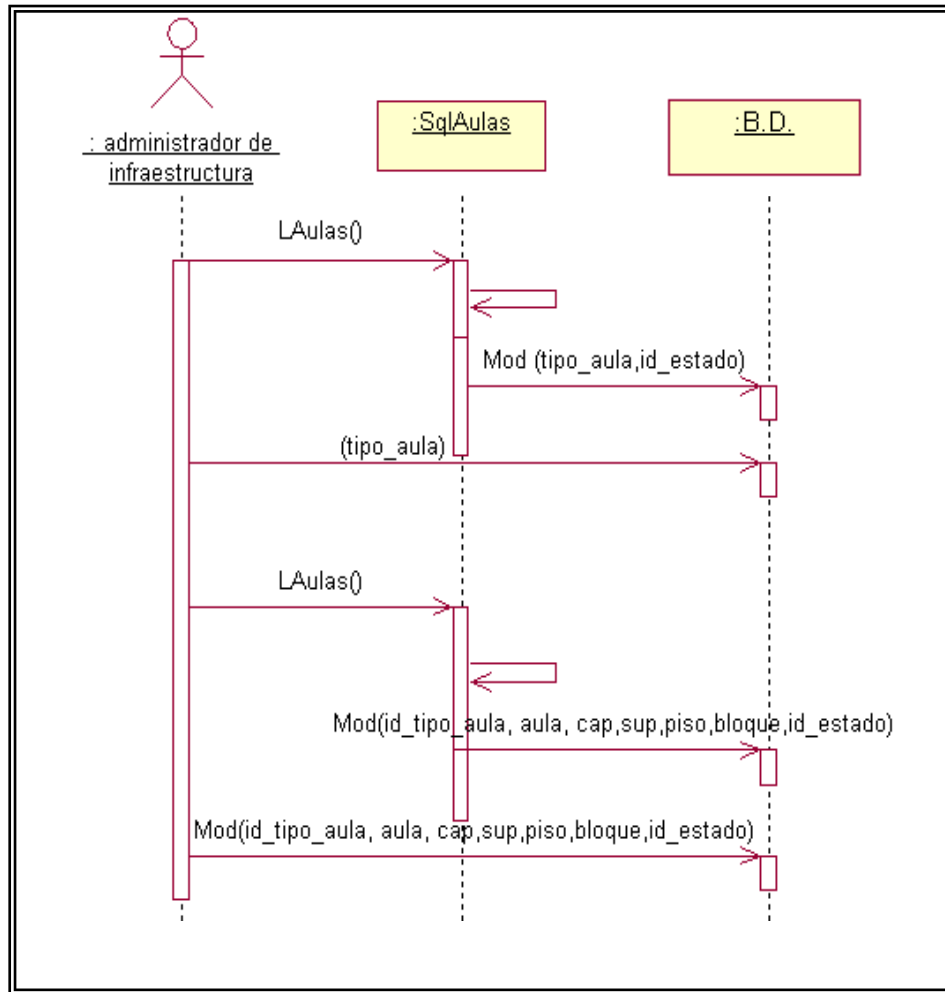


Fig. 3.6.3.1: Diagrama de secuencia Administrar tipos aulas y aulas
Fuente. Elaboración propia

3.6.4. Diagrama de secuencia desde punto de vista del Administrador Académico

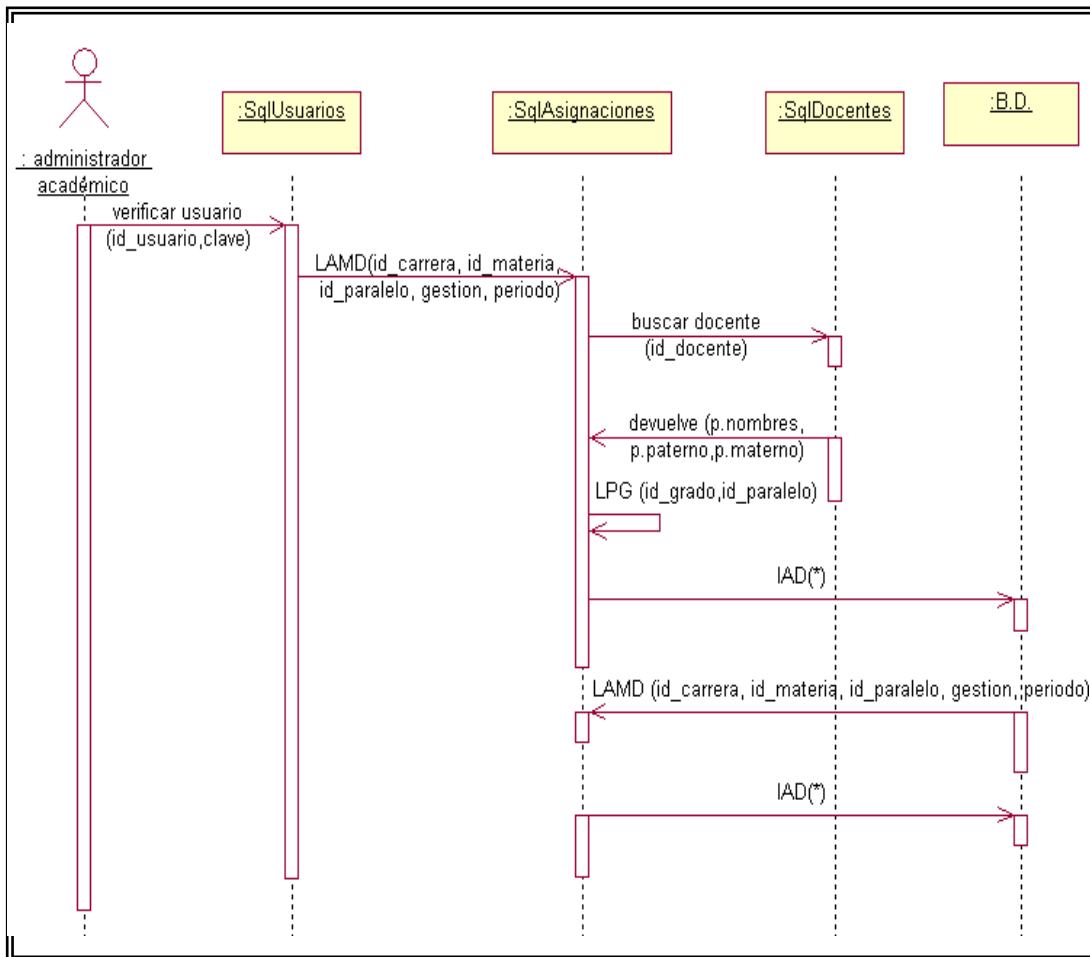


Fig. 3.6.4.1: Diagrama de secuencia Asignación docente

Fuente. Elaboración propia

Parámetro IAD(*): id_docente, id_carrera, id_materia, id_paralelo, id_fase, id_tipo_docente, id_tipo_asignacion, gestion, periodo, observacion, fec_inicio, fec_fin, nro_resolucion, fec_resolucion, id_rol)

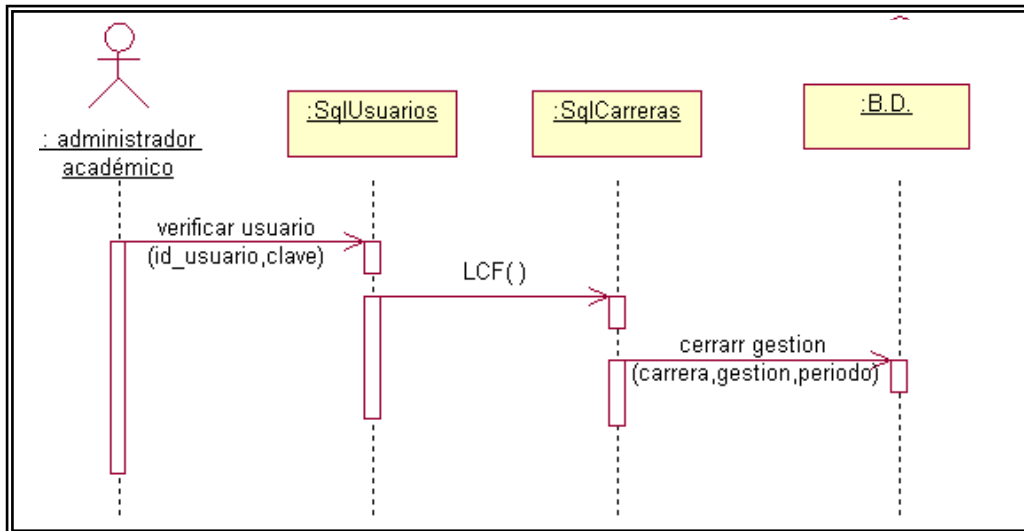


Fig. 3.6.4.2: Diagrama de secuencia Cerrar Gestión

Fuente. Elaboración propia

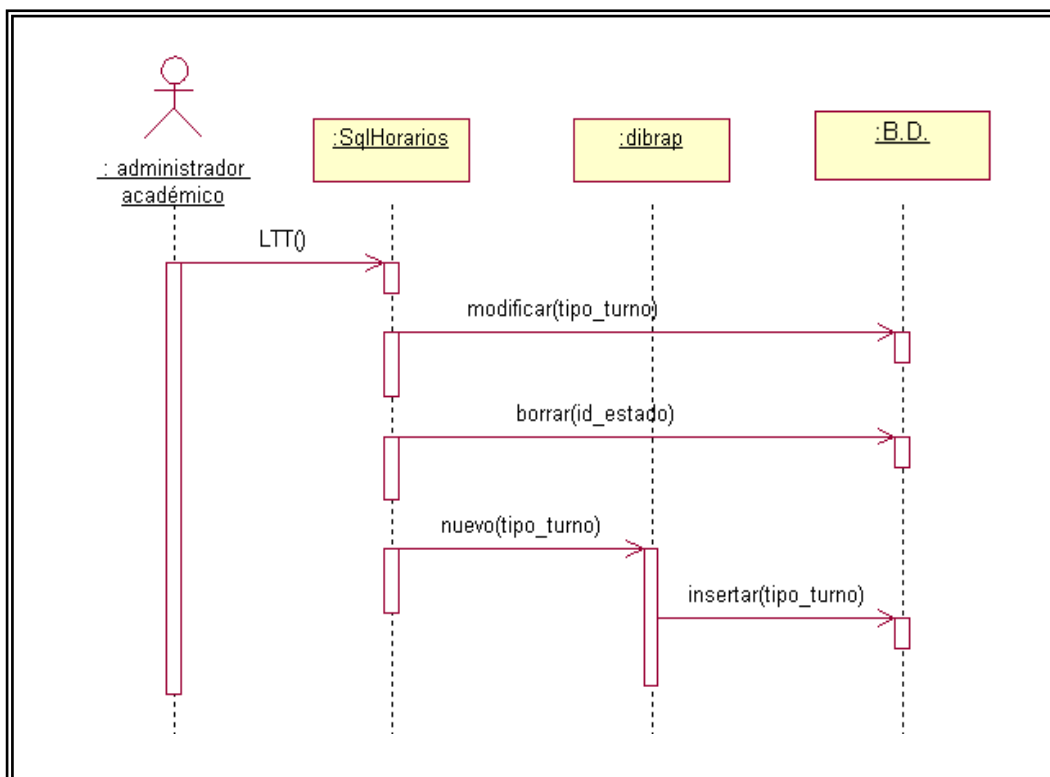


Fig. 3.6.4.3: Diagrama de secuencia Horario Tipo Turnos

Fuente. Elaboración propia

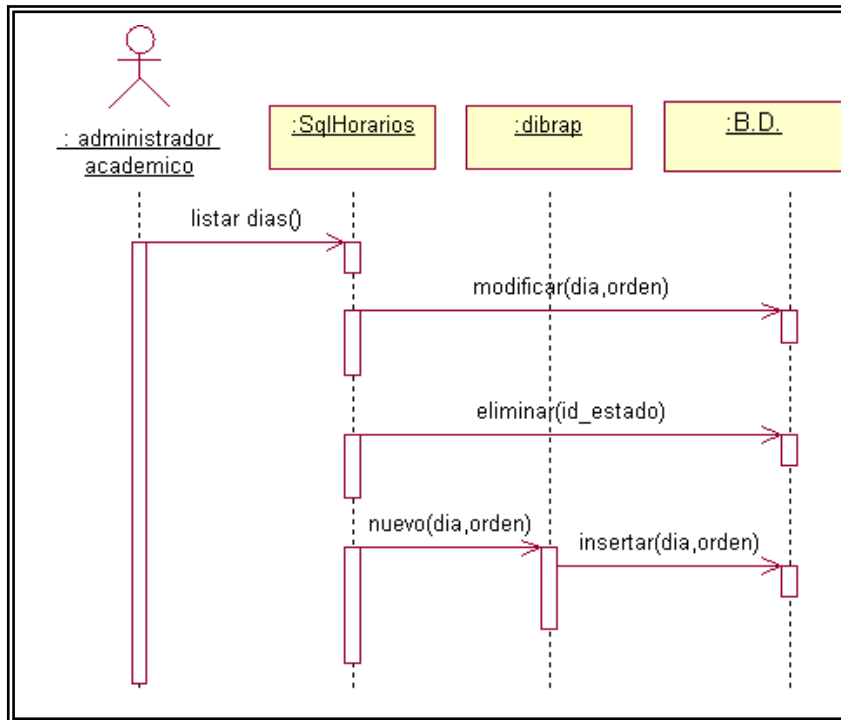


Fig. 3.6.4.4: Diagrama de secuencia administrar días
Fuente. Elaboración propia

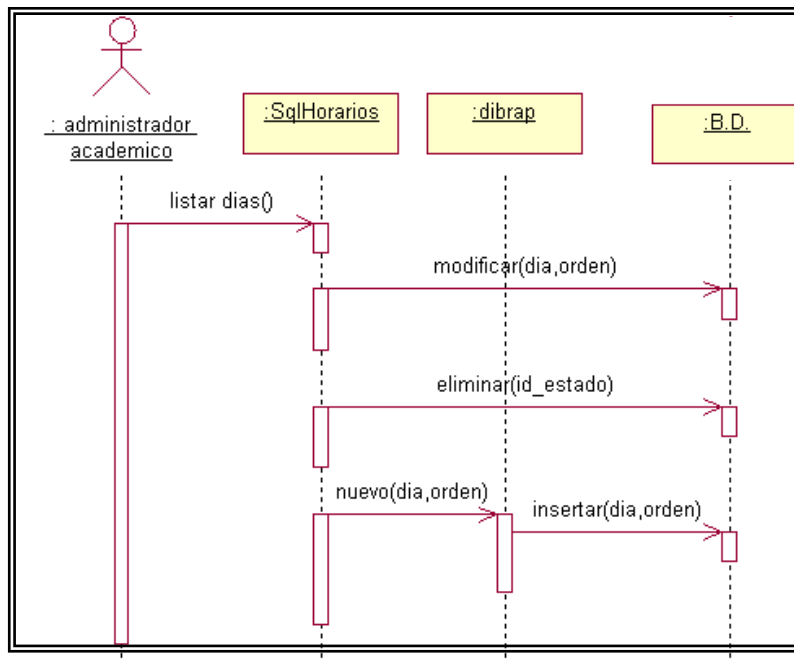


Fig. 3.6.4.5: Diagrama de secuencia Administrar horas
Fuente. Elaboración propia

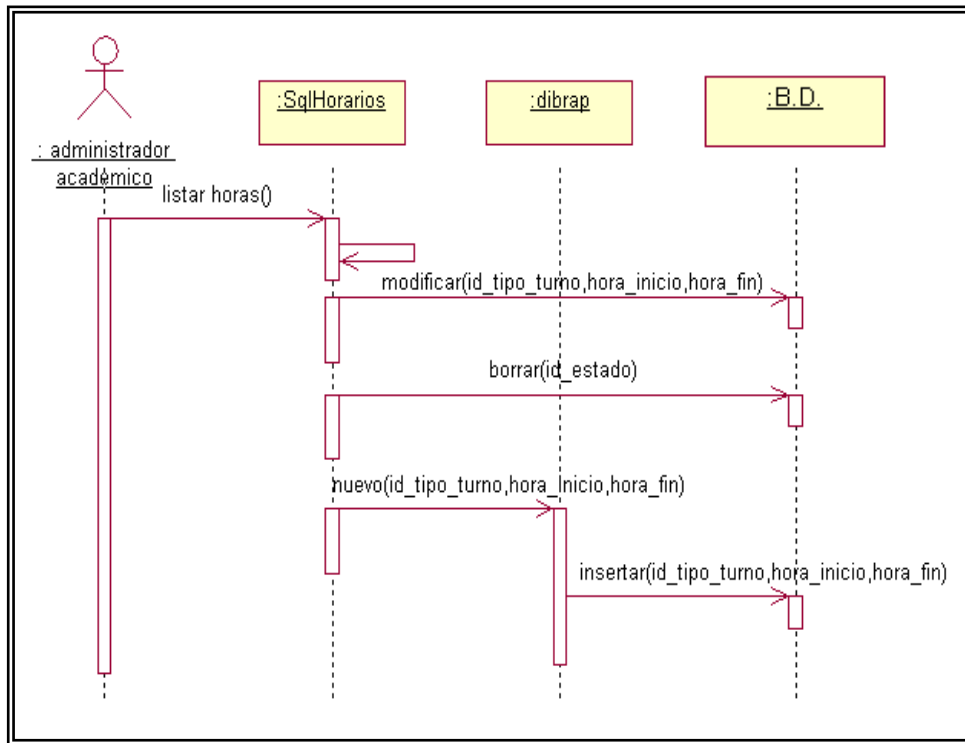


Fig. 3.6.4.6: Diagrama de secuencia administrar horarios
Fuente. Elaboración propia

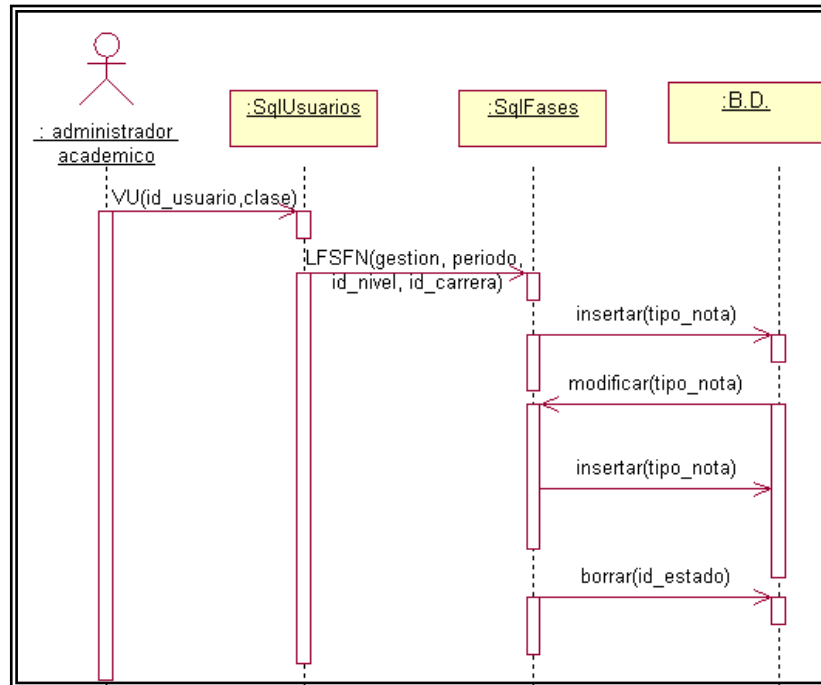


Fig. 3.6.4.7: Diagrama de secuencia Administrar tipo notas
Fuente. Elaboración propia

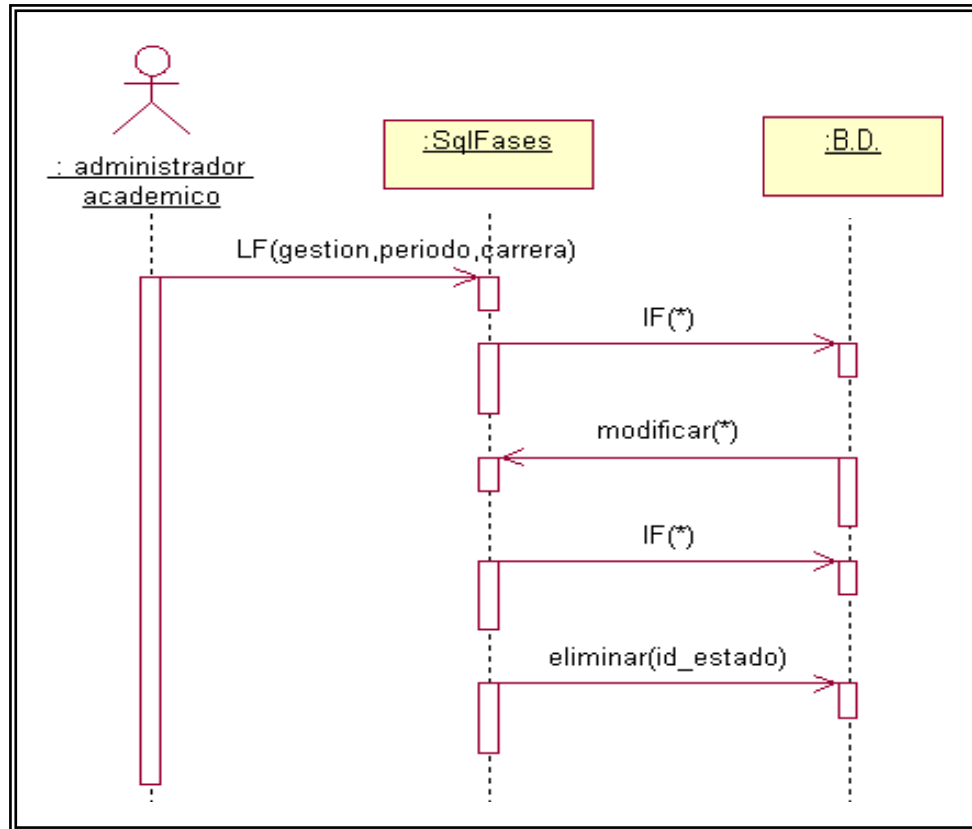


Fig. 3.6.4.8: Diagrama de secuencia administrar fases
Fuente. Elaboración propia

Parametros IF(), modificar(*):**id_fase,
id_carrera, id_nivel, gestion, periodo, fase,
ponderacion, nota_maxima, nota_aprobacion,
descripcion. id rol. id usuario

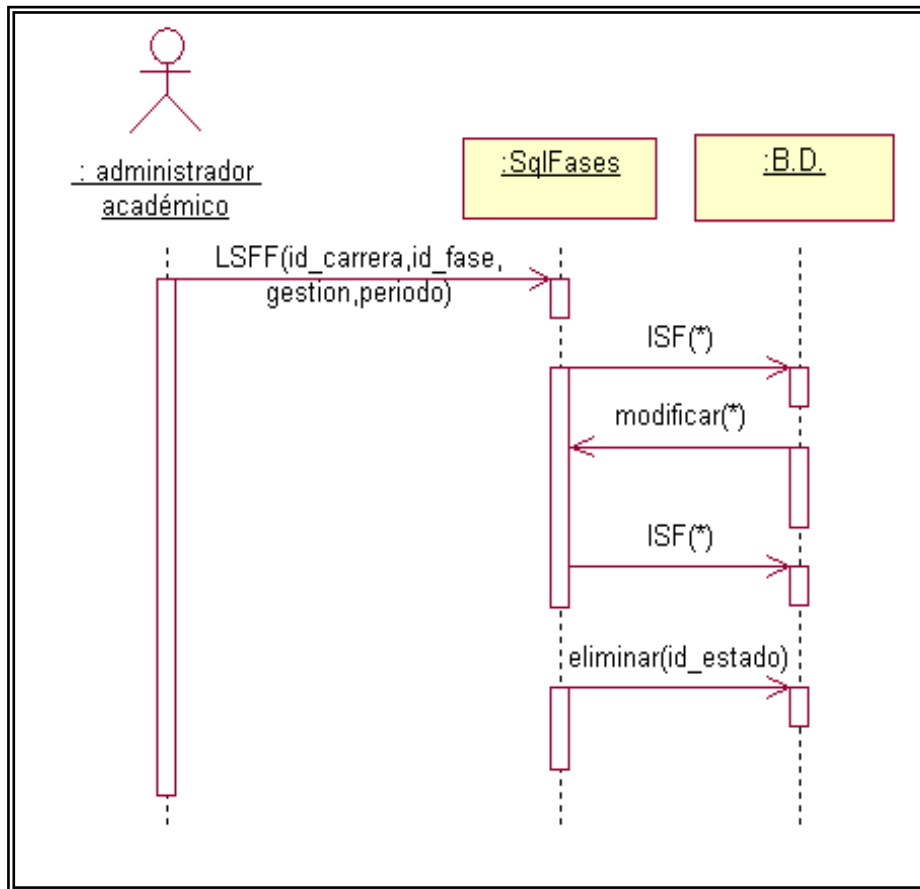


Fig. 3.6.4.9: Diagrama de secuencia administrar subfases
Fuente. Elaboración propia

Parámetros IF(*), Modificar():**

id_fase, id_carrera, id_nivel, gestion,
 periodo, fase, ponderacion, nota_maxima,
 nota_aprobacion, descripcion, id_rol,
 id usuario

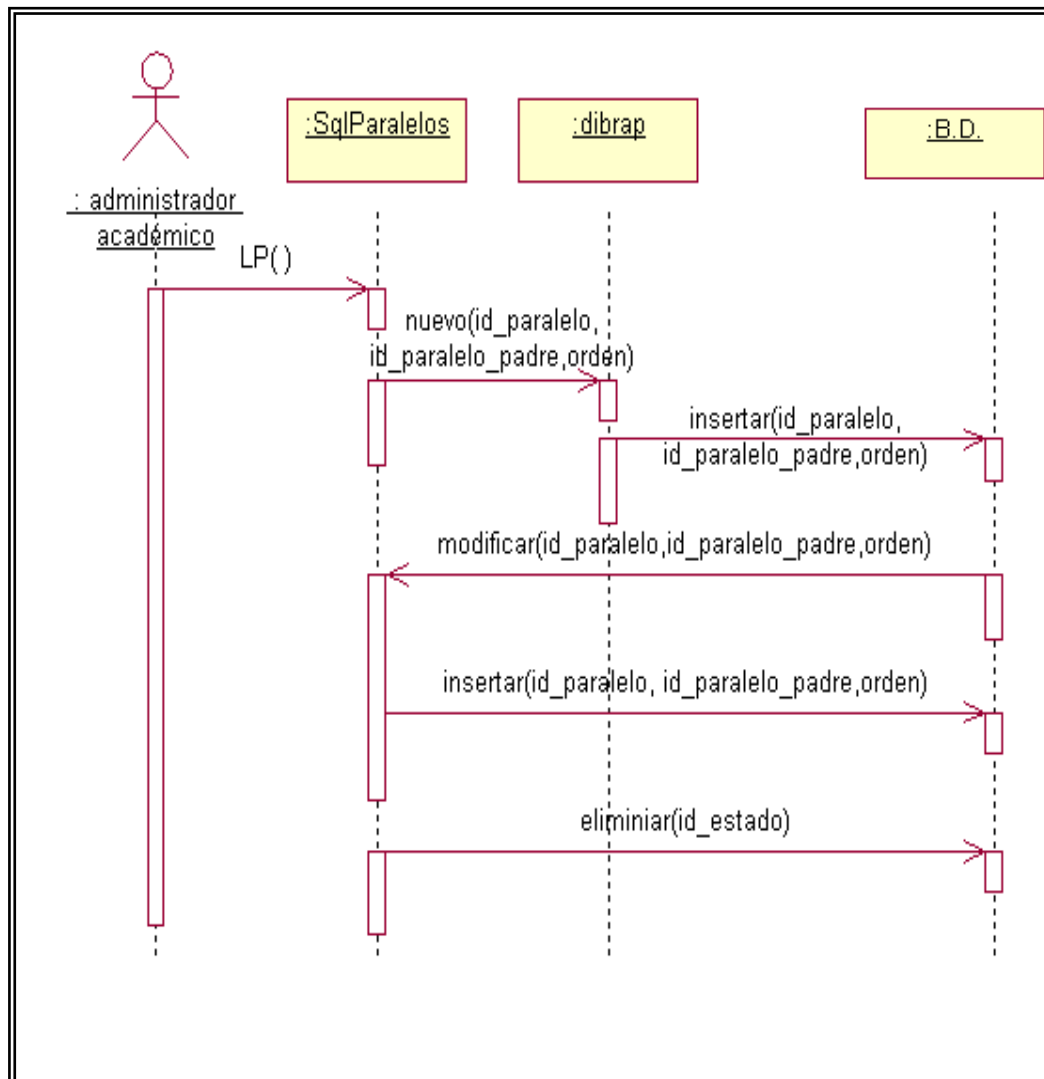


Fig. 3.6.4.10: Diagrama de secuencia administrar paralelos

Fuente. Elaboración propia

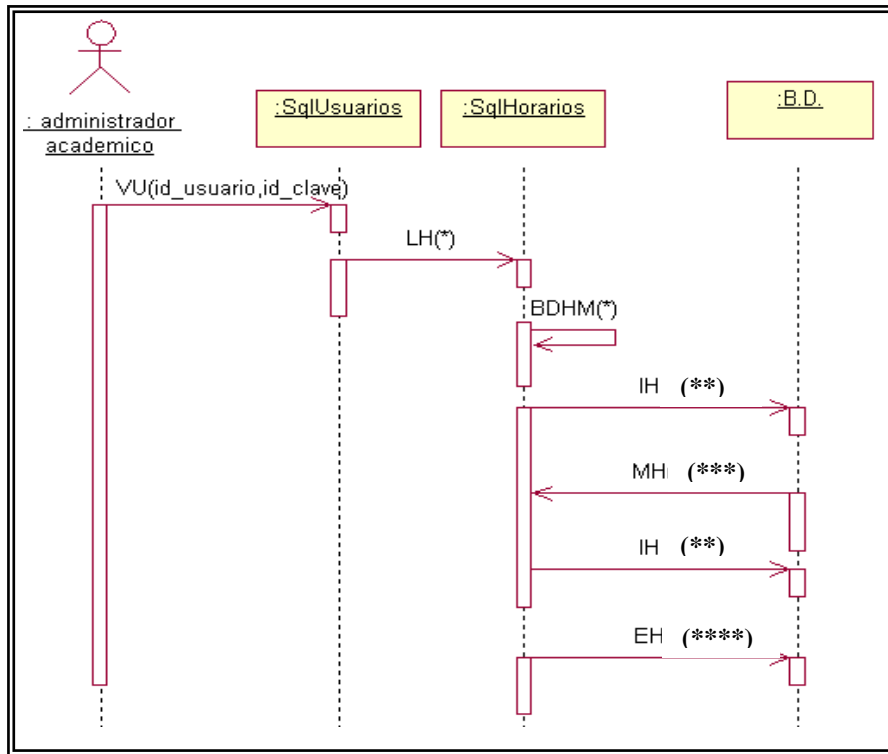


Fig. 3.6.4.11: Diagrama de secuencia administrar horarios
Fuente. Elaboración propia

Parámetros LH(*)id_carrera,
 id_plan, id_aula, id_grado,
 id_paralelo, gestion, periodo,
 id_dia, id_hora)

ParámetrosEH (**)**
 id_horario, id_paralelo,
 id_materia, gestion, periodo,
 id_dia ,id_hora, ult_usuario

Parámetros IH()**id_carrera,
 id_plan, id_grado, id_materia,
 id_paralelo, id_dia, id_hora, id_aula,
 gestion, periodo

Parámetros MH(*)**id_grado,
 id_paralelo, id_materia,
 gestion, periodo, id_dia
 ,id_hora, ult_usuario

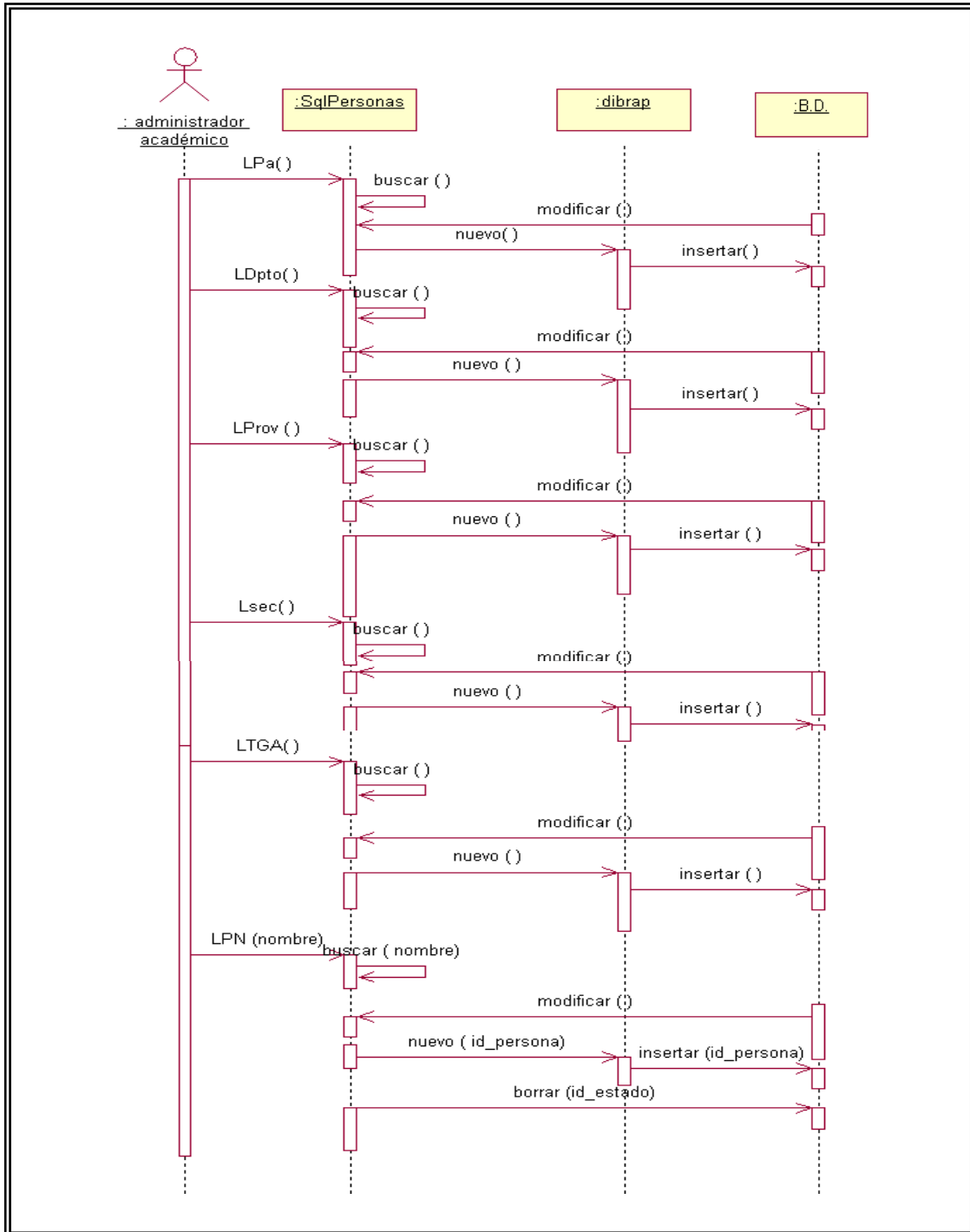


Fig. 3.6.4.12: Diagrama de secuencia administrar personas
Fuente. Elaboración propia

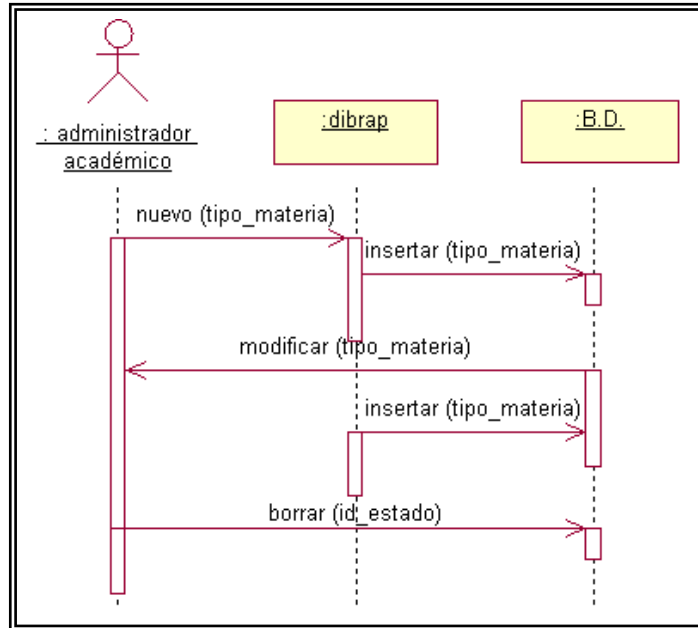


Fig. 3.6.4.13: Diagrama de secuencia administrar tipo materias
Fuente. Elaboración propia

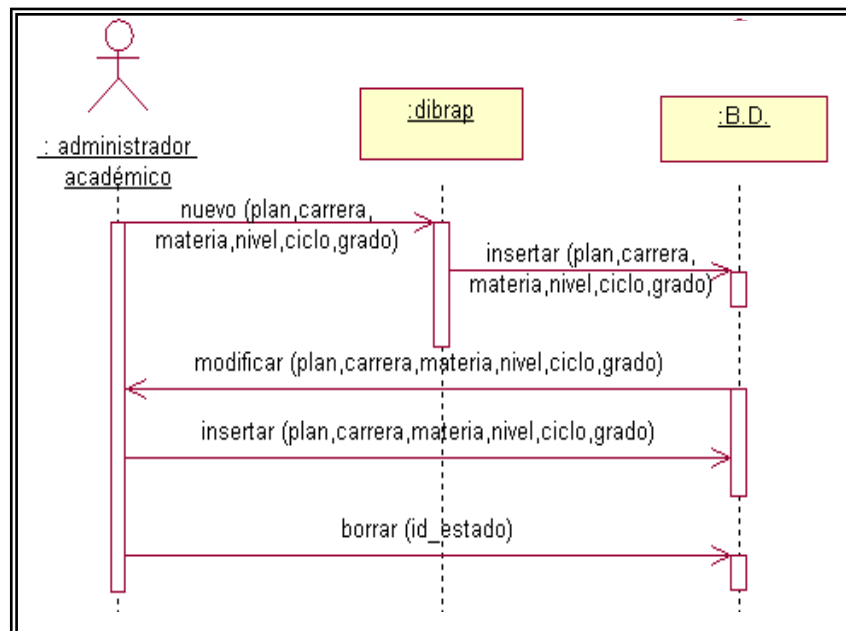


Fig. 3.6.4.14: Diagrama de secuencia administrar planes de estudio
Fuente. Elaboración propia

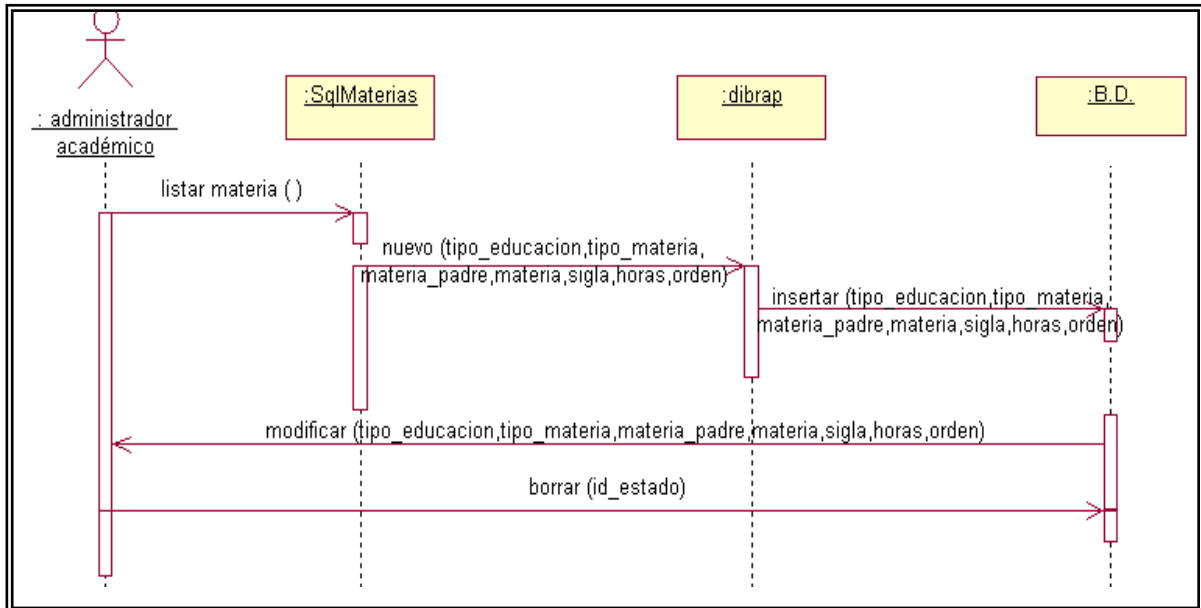


Fig. 3.6.15: Diagrama de secuencia administrar materias
Fuente. Elaboración propia

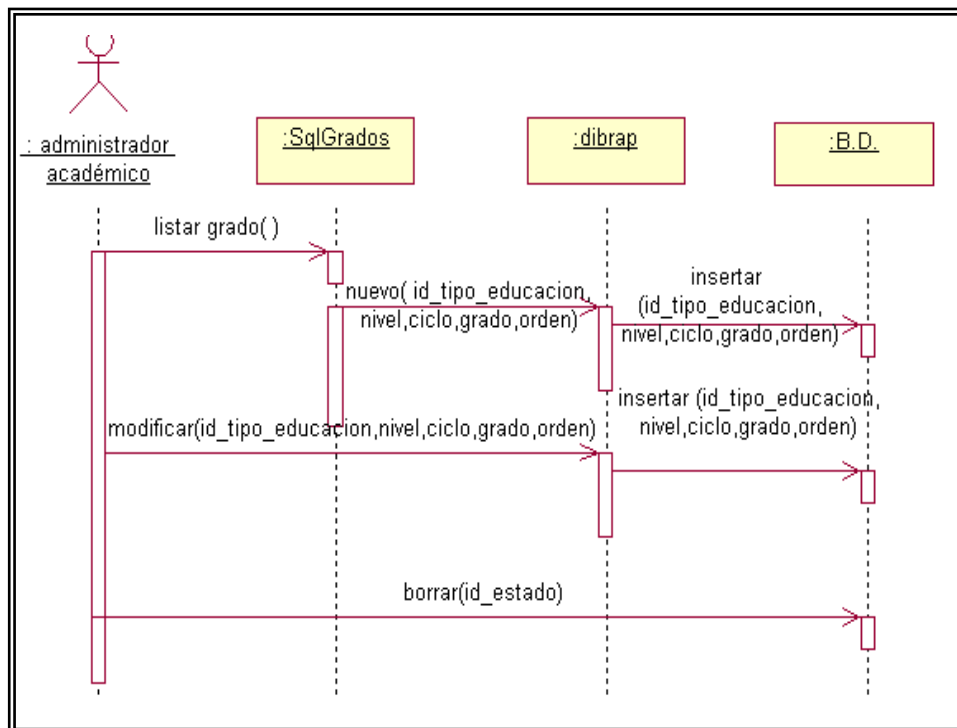


Fig. 3.6.4.16: Diagrama de secuencia administrar grados
Fuente. Elaboración propia

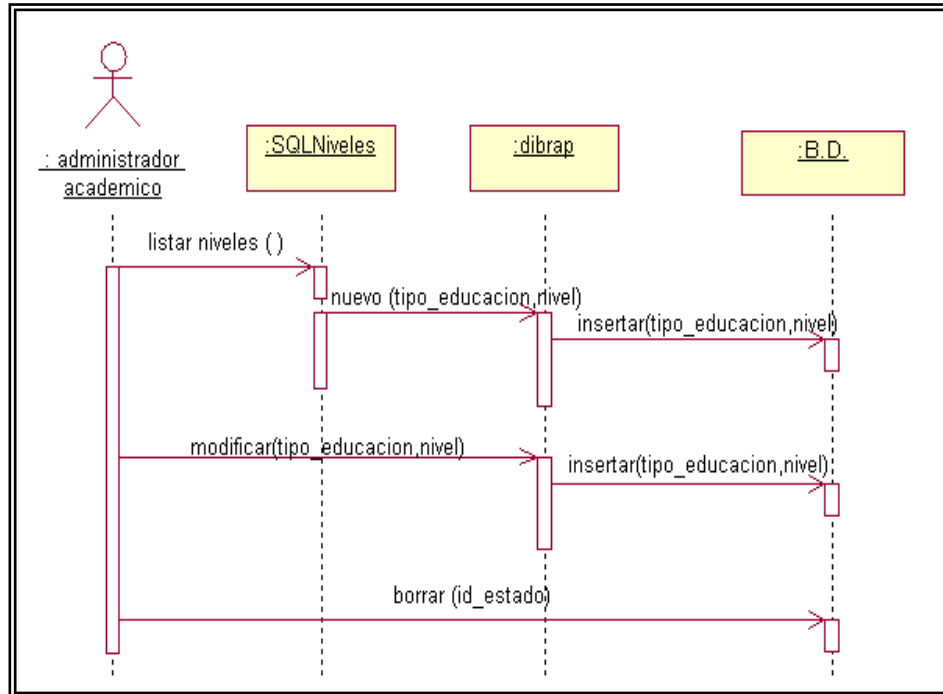


Fig. 3.6.4.17: Diagrama de secuencia administrar niveles
Fuente. Elaboración propia

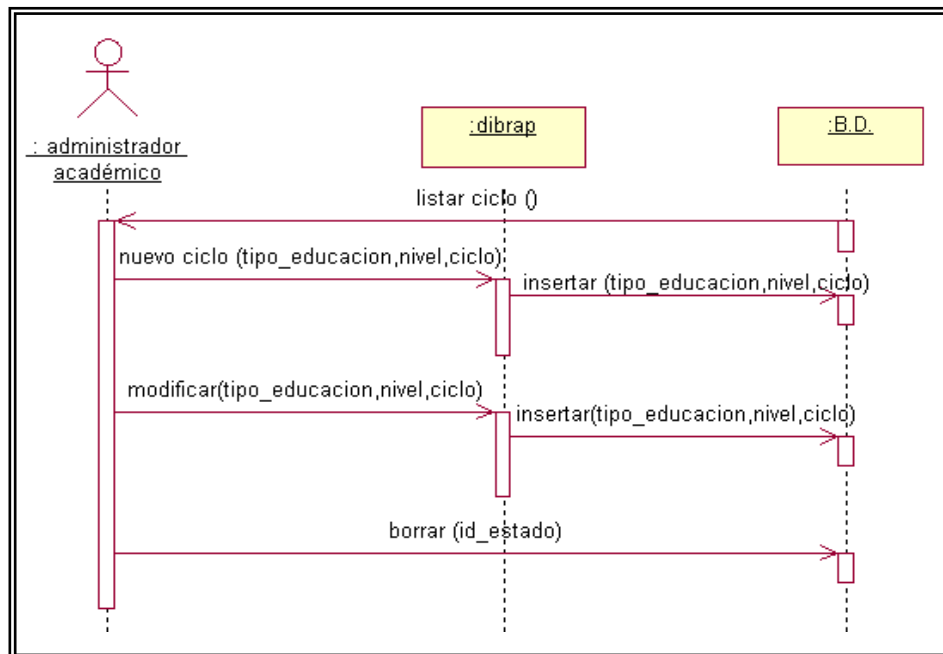


Fig. 3.6.4.18: Diagrama de secuencia administrar ciclos
Fuente. Elaboración propia

3.6.5. Diagrama de secuencia desde punto de vista del Administrador de Registro Humanístico.

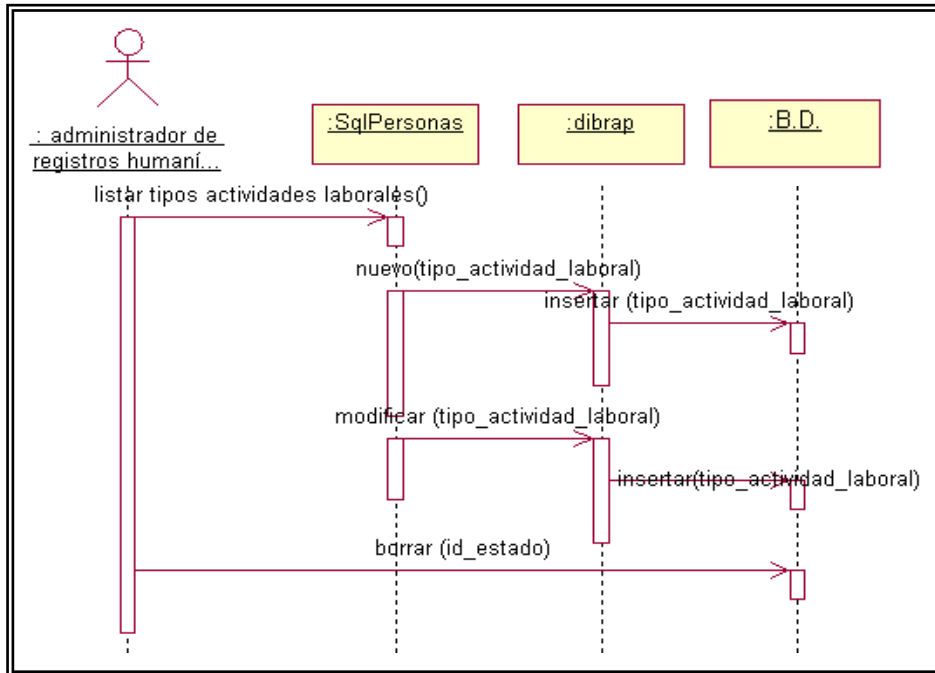


Fig. 3.6.5.1: Diagrama de secuencia tipo de actividad Laboral
Fuente. Elaboración propia

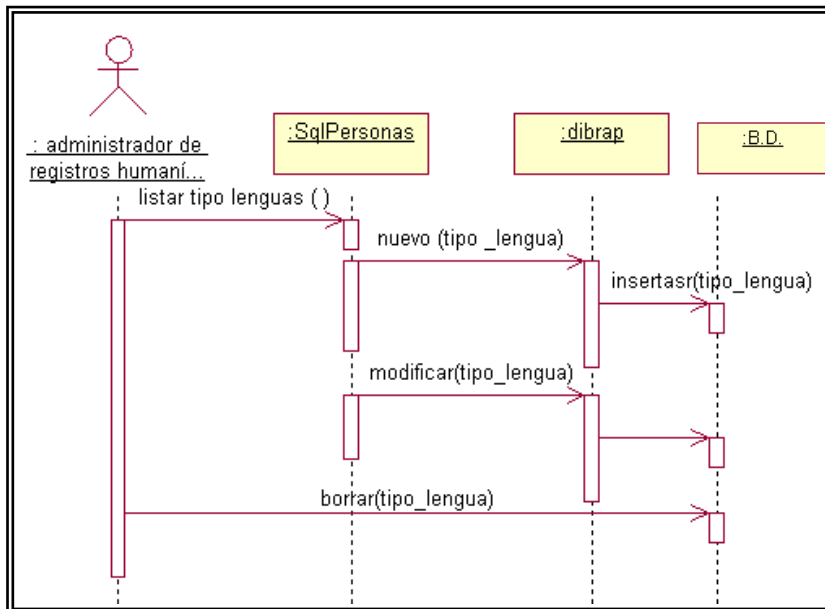


Fig. 3.6.5.2: Diagrama de secuencia administrar tipo lengua originaria
Fuente. Elaboración propia

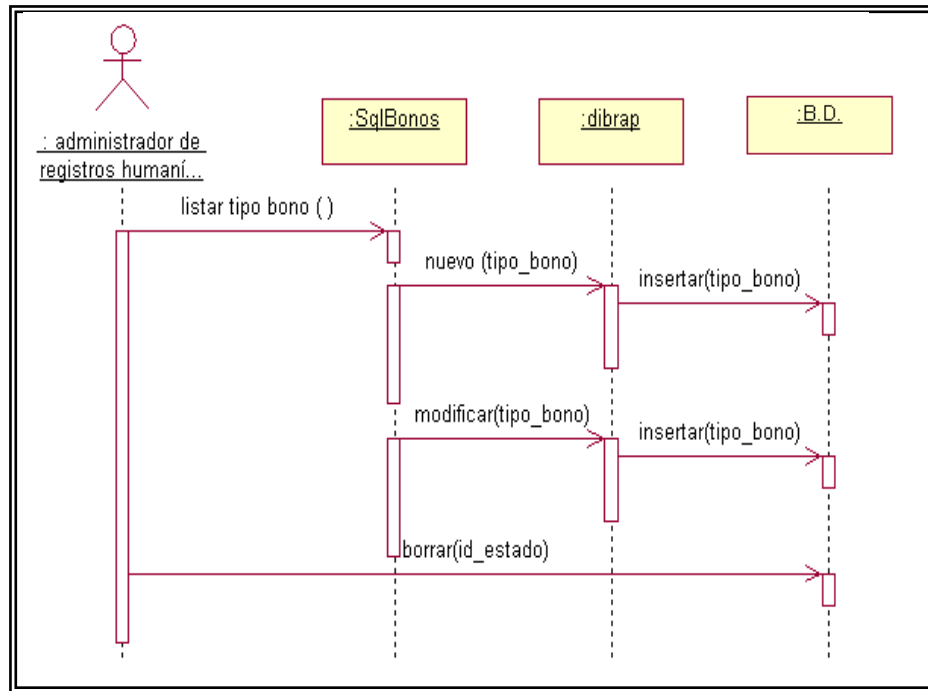


Fig. 3.6.5.3: Diagrama de secuencia administrar tipo bono
Fuente. Elaboración propia

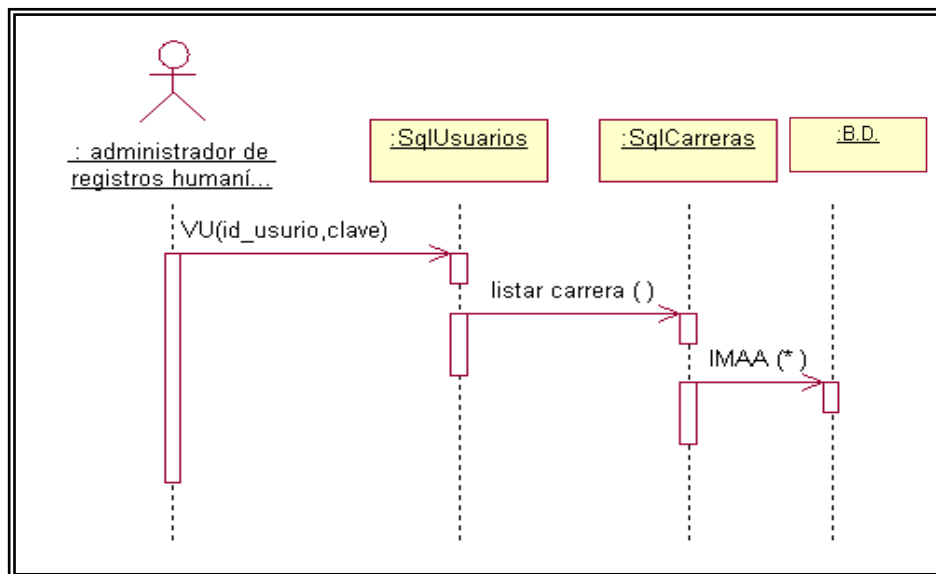


Fig. 3.6.5.4: Diagrama de secuencia inscripción de estudiante antiguo
Fuente. Elaboración propia

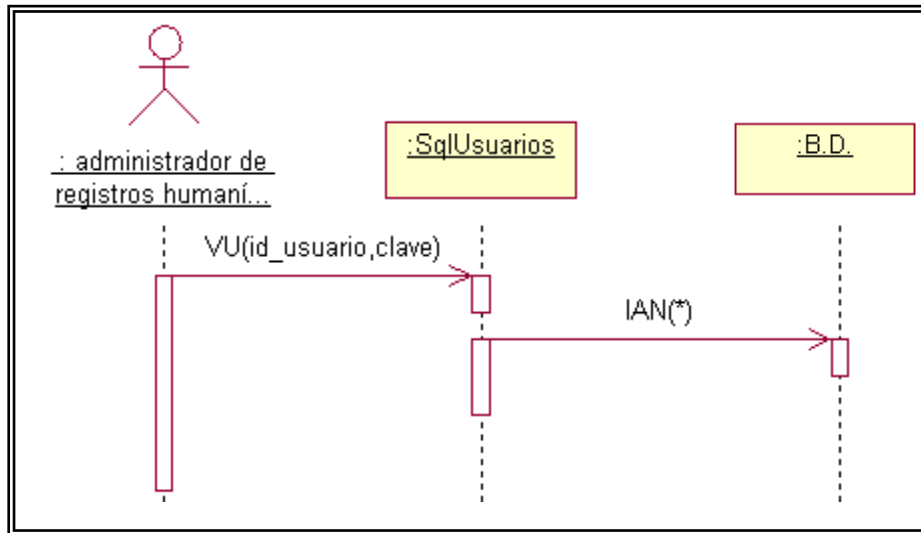


Fig. 3.6.5.5: Diagrama de secuencia inscripción alumno nuevo
Fuente. Elaboración propia

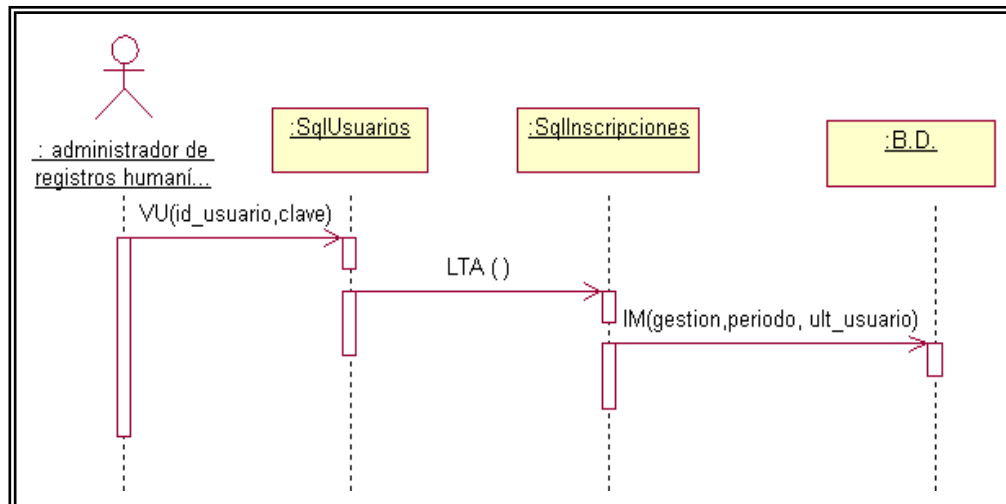


Fig. 3.6.5.6: Diagrama de secuencia inscripción de estudiante por traspaso
Fuente. Elaboración propia

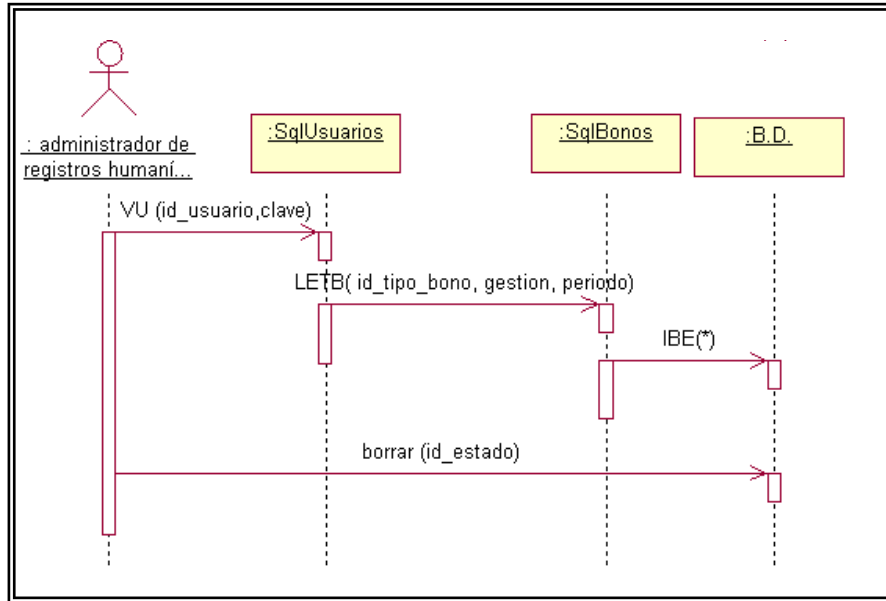


Fig. 3.6.5.7: Diagrama de secuencia administrar bono estudiante
Fuente. Elaboración propia

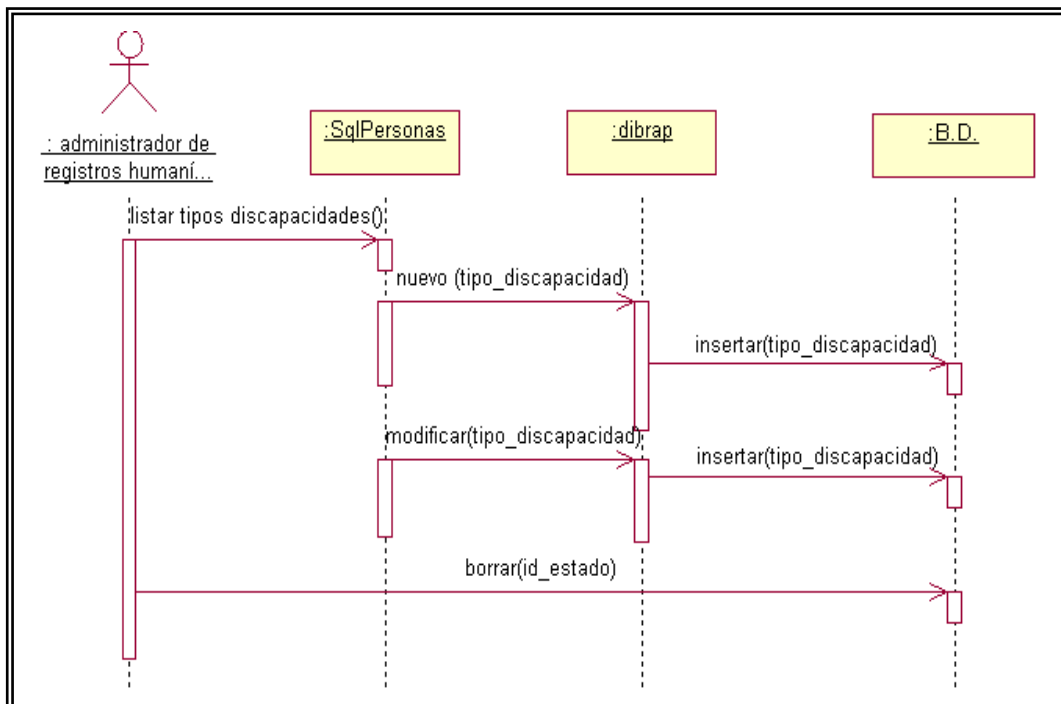


Fig. 3.6.5.8: Diagrama de secuencia administrar Tipo discapacidades
Fuente. Elaboración propia

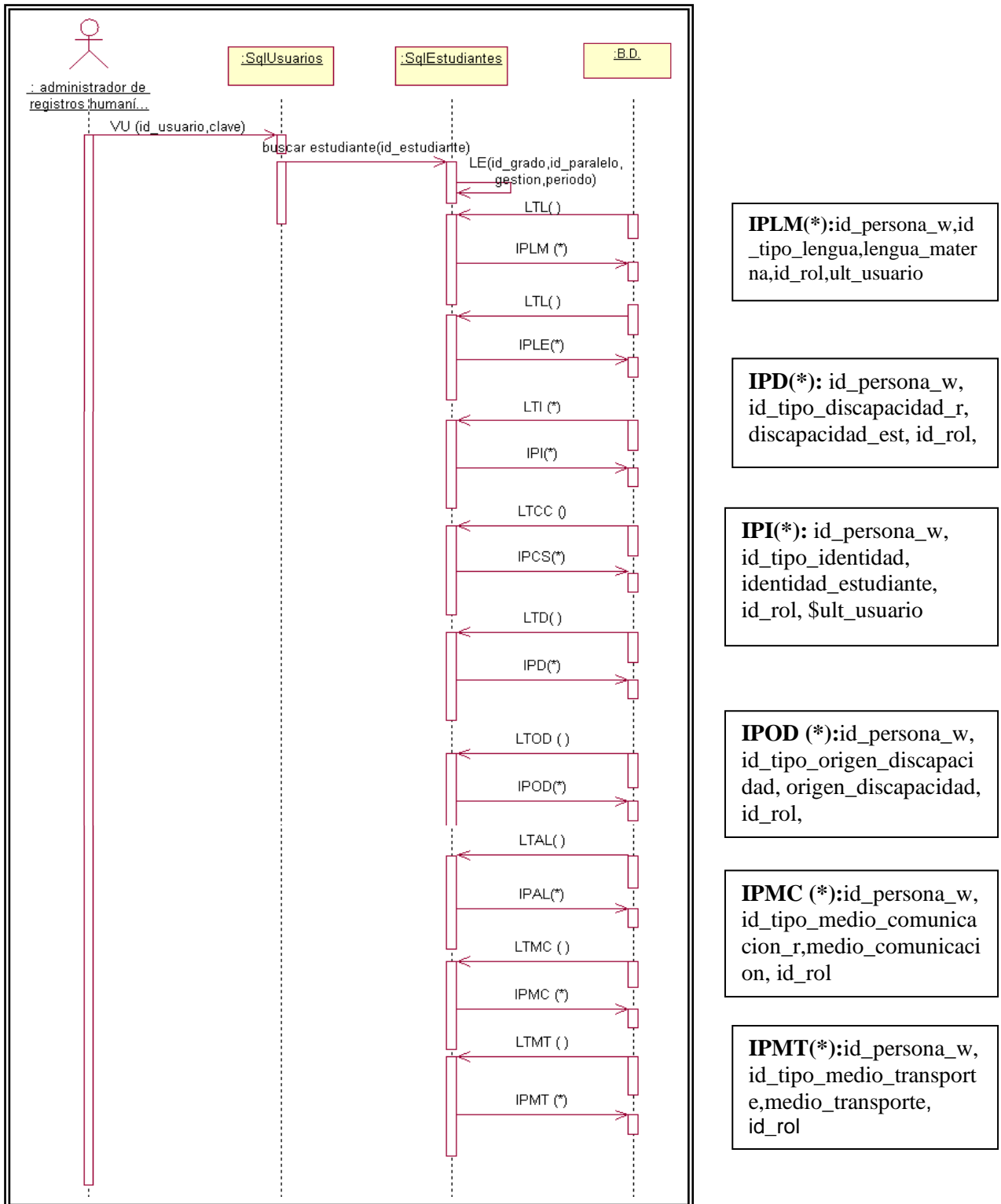
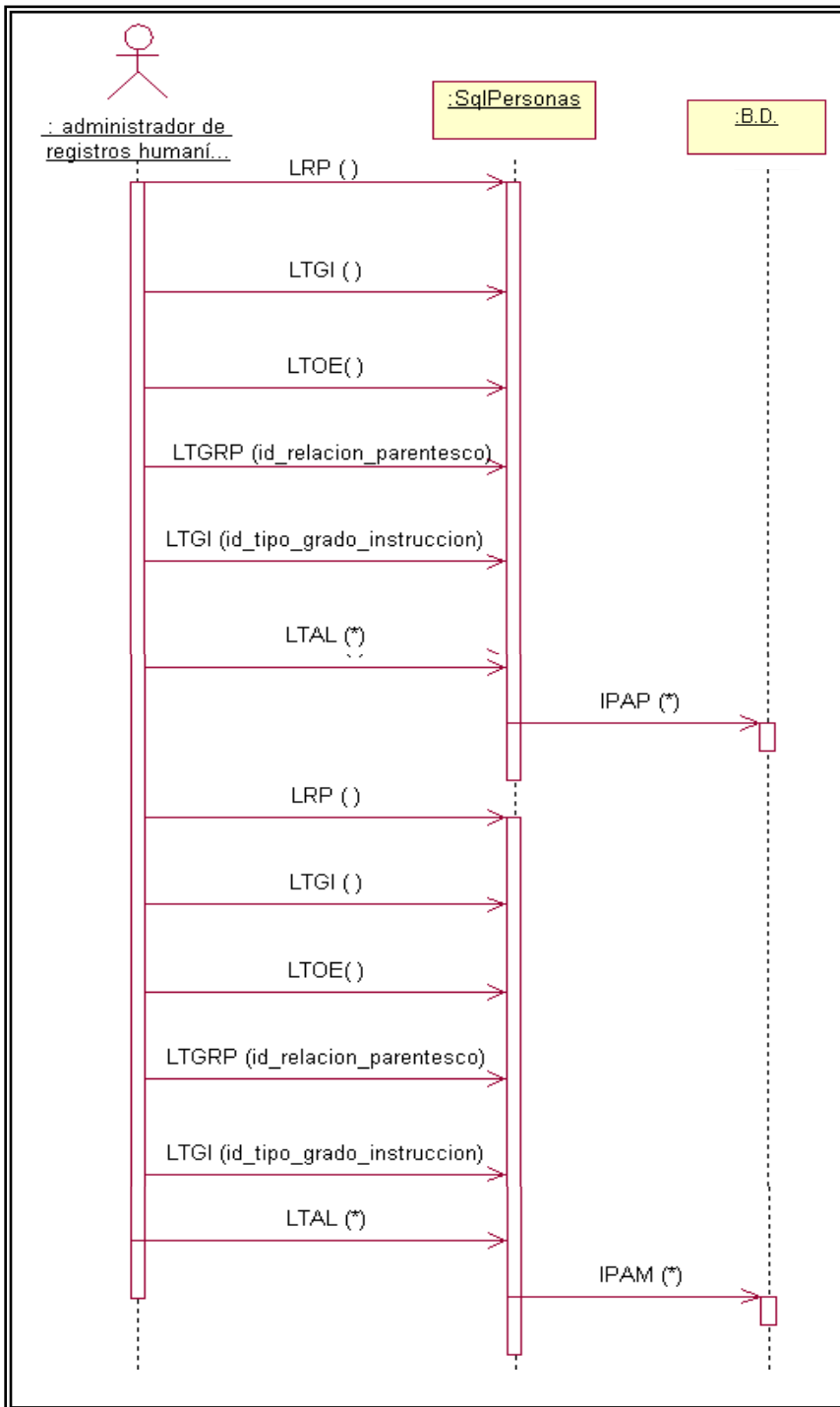


Fig. 3.6.5.9: Diagrama de secuencia registro de encuesta social
Fuente. Elaboración propia



IPAP (*):
 id_persona_w, id_relacion_pa
 rentesco_p,
 id_tipo_actividad_laboral_p,
 id_tipo_grado_instruccion_p,
 id_tipo_origen_etnico_p,
 id_tipo_lengua_p,
 nombres_p, paterno_p,
 materno_p, cedula_p,

IPAM (*):
 id_persona_w, id_relacion_pa
 rentesco_p,
 id_tipo_actividad_laboral_p,
 id_tipo_grado_instruccion_p,
 id_tipo_origen_etnico_p,
 id_tipo_lengua_p,
 nombres_p, paterno_p,
 materno_p, cedula_p,

Fig. 3.6.5.10: Diagrama de secuencia registro tutores
Fuente. Elaboración propia

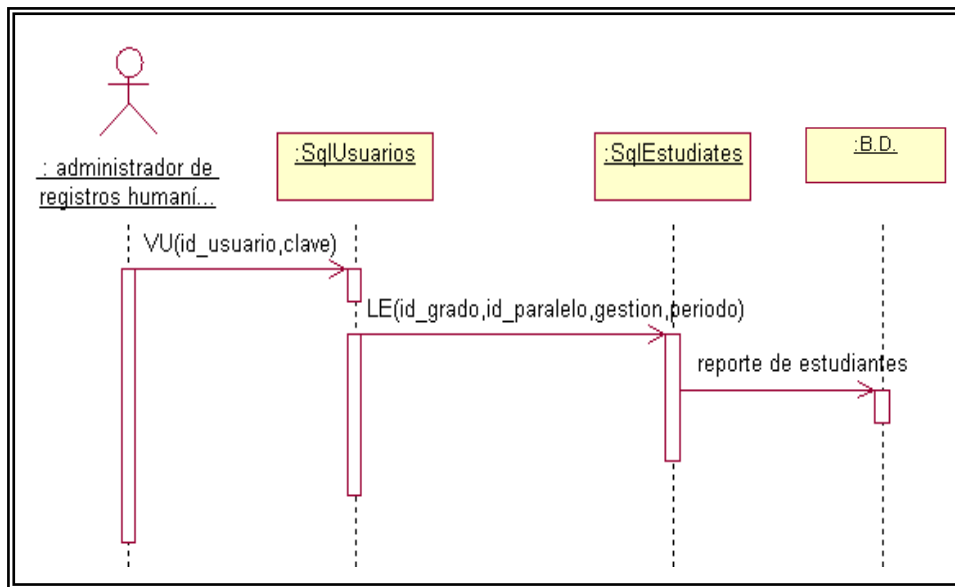


Fig. 3.6.5.11: Diagrama de secuencia Lista de estudiantes por paralelo
Fuente. Elaboración propia

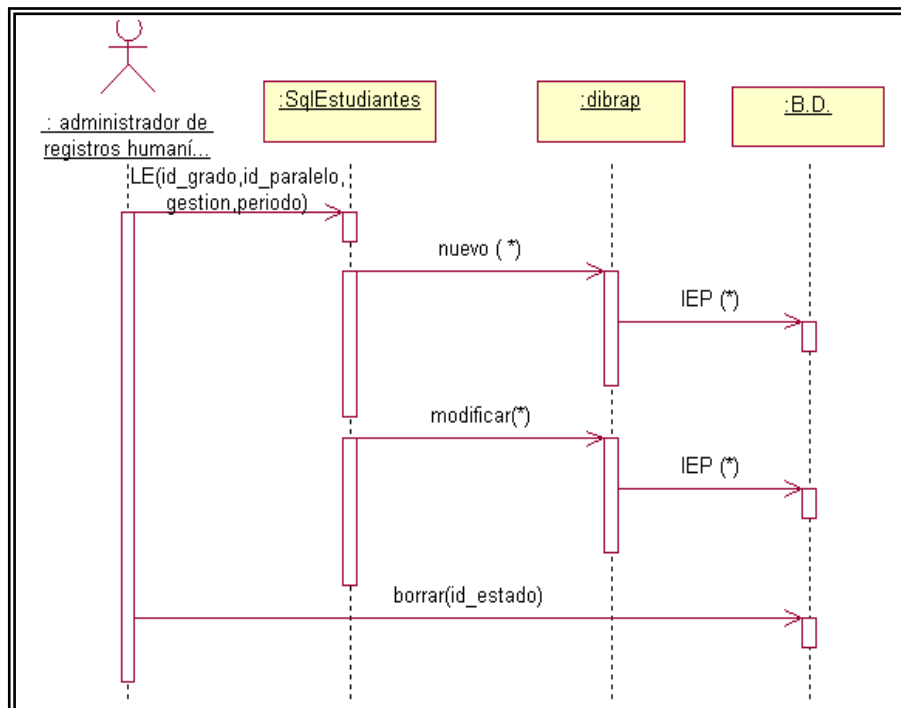


Fig. 3.6.5.12: Diagrama de secuencia plan de estudiantes
Fuente. Elaboración propia

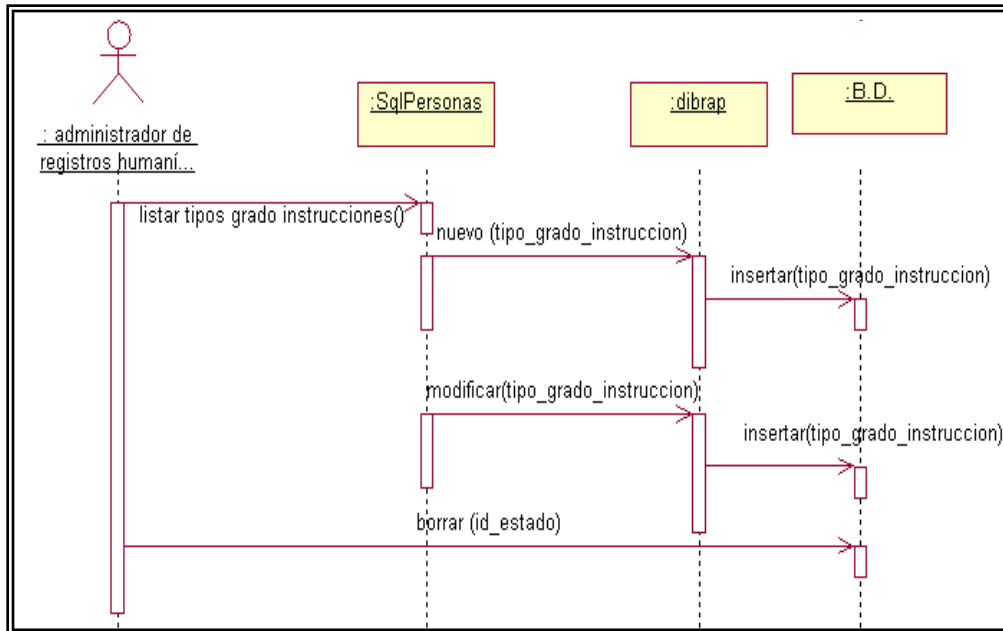


Fig. 3.6.5.13: Diagrama de secuencia grado instrucción tutor
Fuente. Elaboración propia

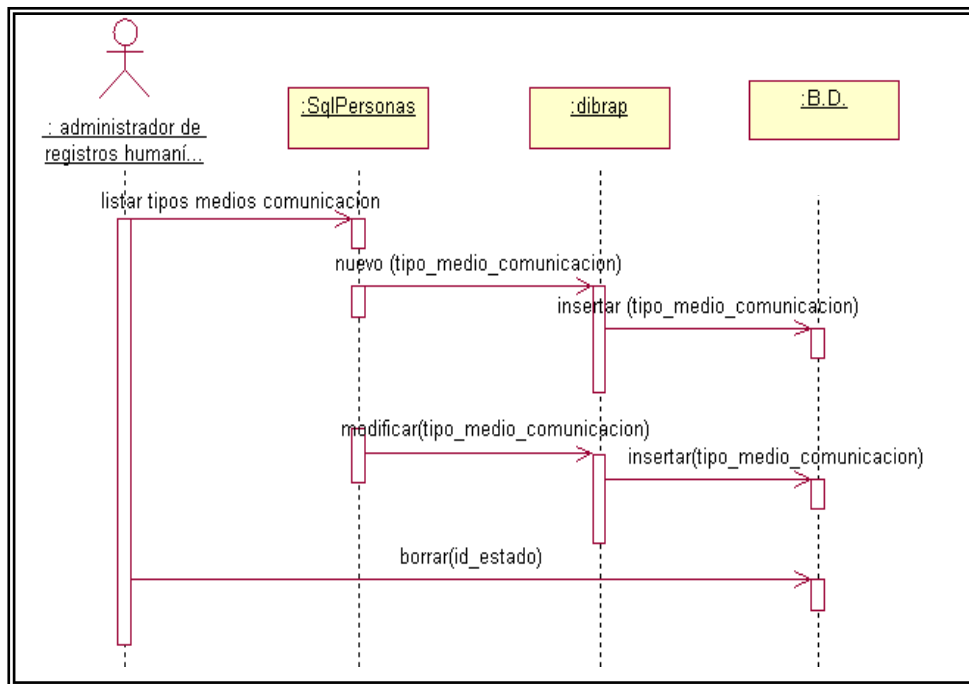


Fig. 3.6.5.14: Diagrama de secuencia registrar medios de comunicación
Fuente. Elaboración propia

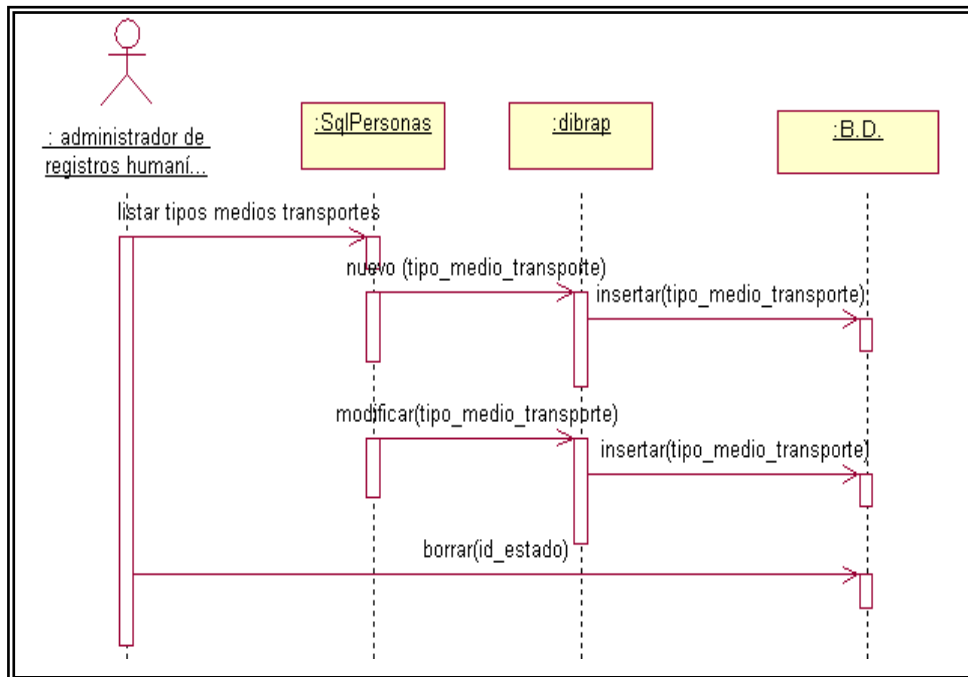


Fig. 3.6.5.15: Diagrama de secuencia registro de medios de transporte
Fuente. Elaboración propia

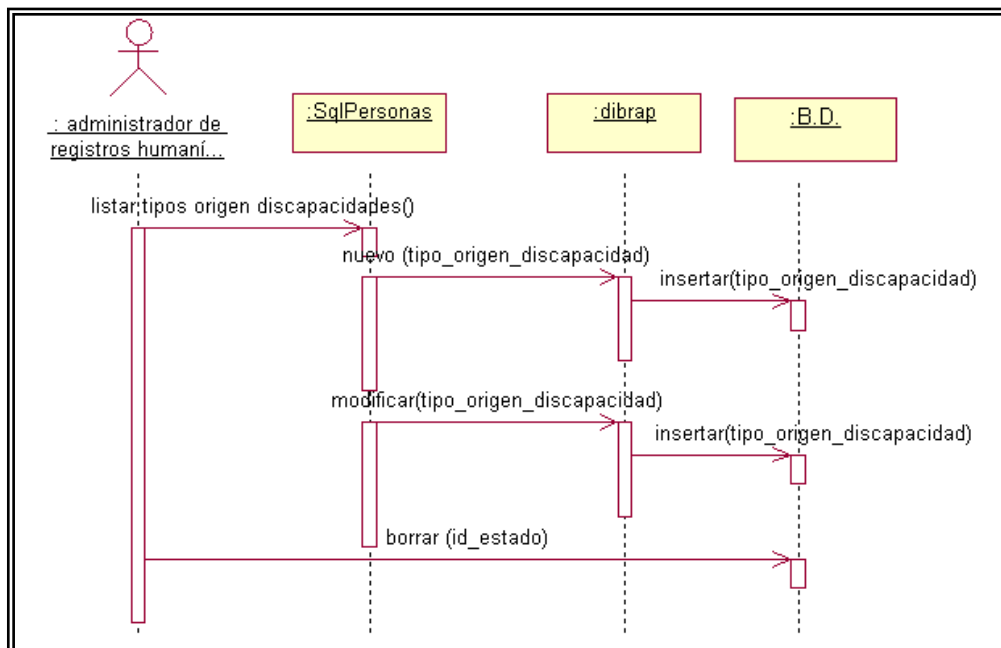


Fig. 3.6.5.16: Diagrama de secuencia registrar origen discapacidades

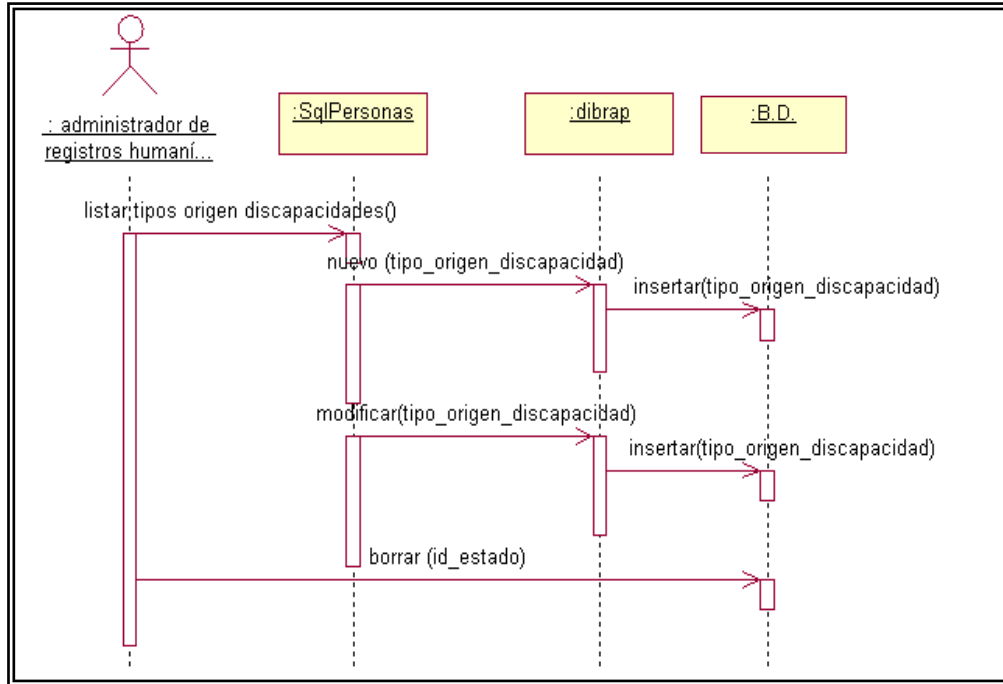


Fig. 3.6.5.17: Diagrama de secuencia registro de discapacidades
Fuente. Elaboración propia

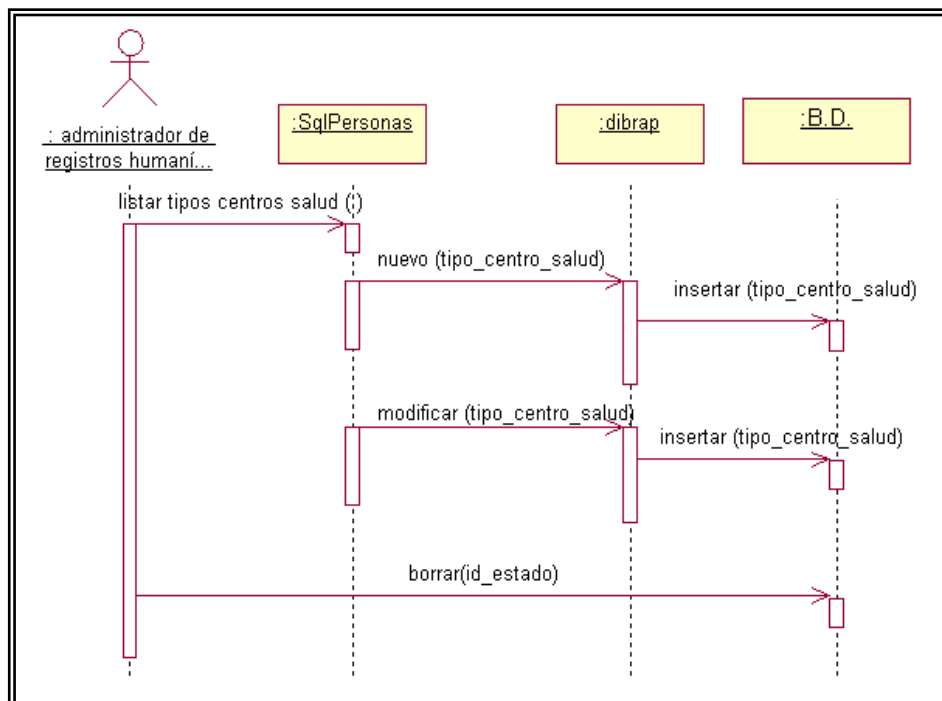


Fig. 3.6.5.18: Diagrama de secuencia registrar centros de salud
Fuente. Elaboración propia

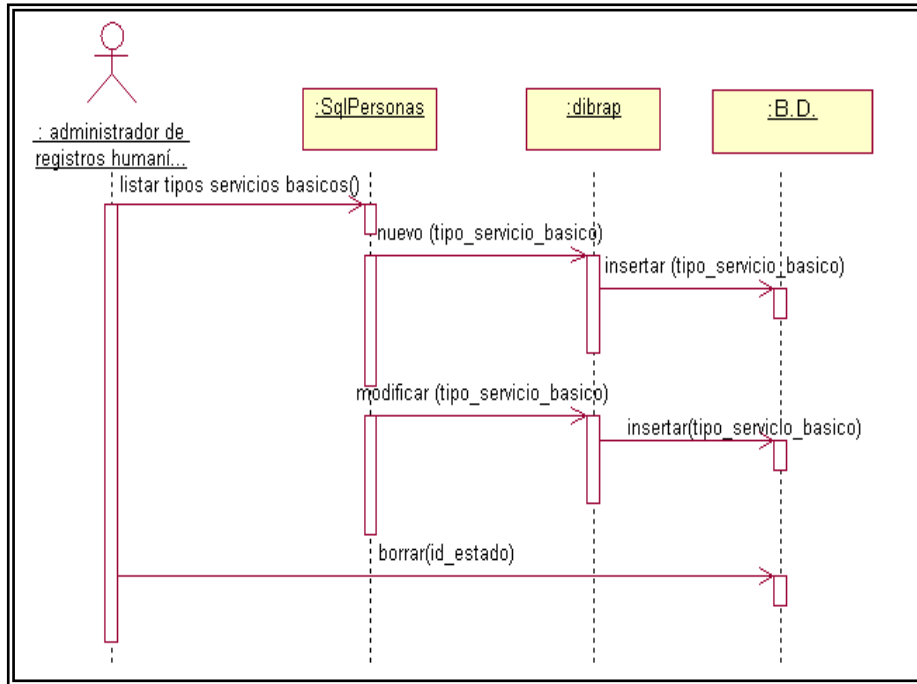


Fig. 3.6.5.19: Diagrama de secuencia registro de servicios básicos

Fuente. Elaboración propia

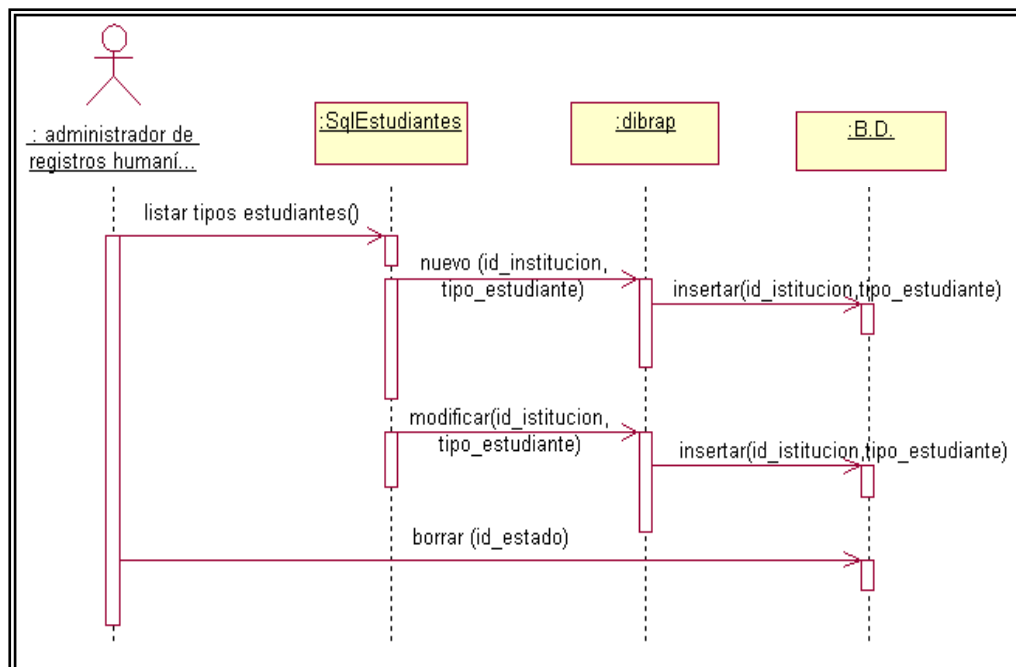


Fig. 3.6.5.20: Diagrama de secuencia registrar tipo de estudiantes

Fuente. Elaboración propia

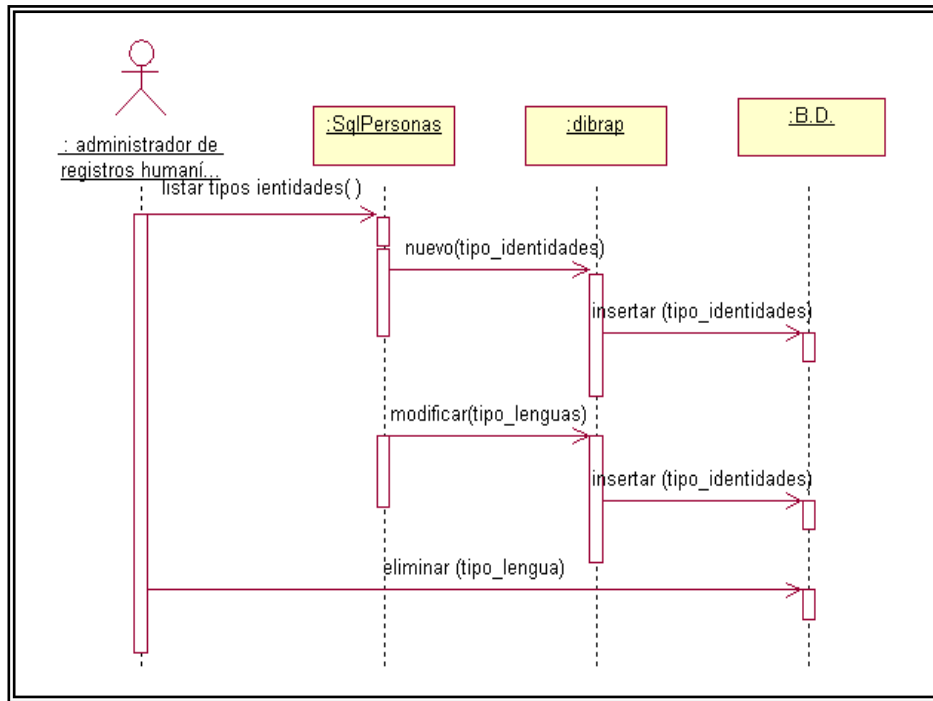


Fig. 3.6.5.21: Diagrama de secuencia registrar tipo de identidades
Fuente. Elaboración propia

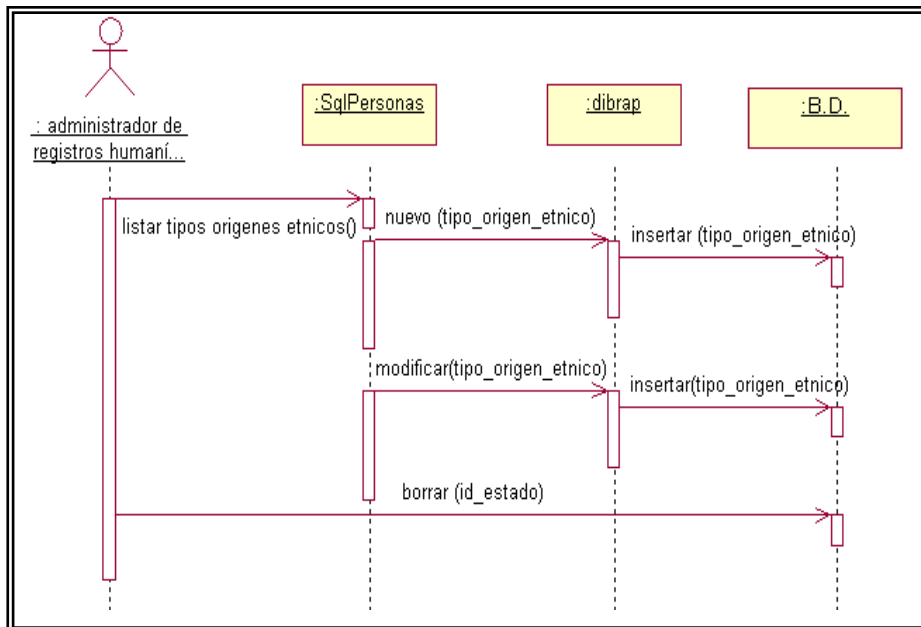


Fig. 3.6.5.22: Diagrama de secuencia registrar tipo de origen étnico
Fuente. Elaboración propia

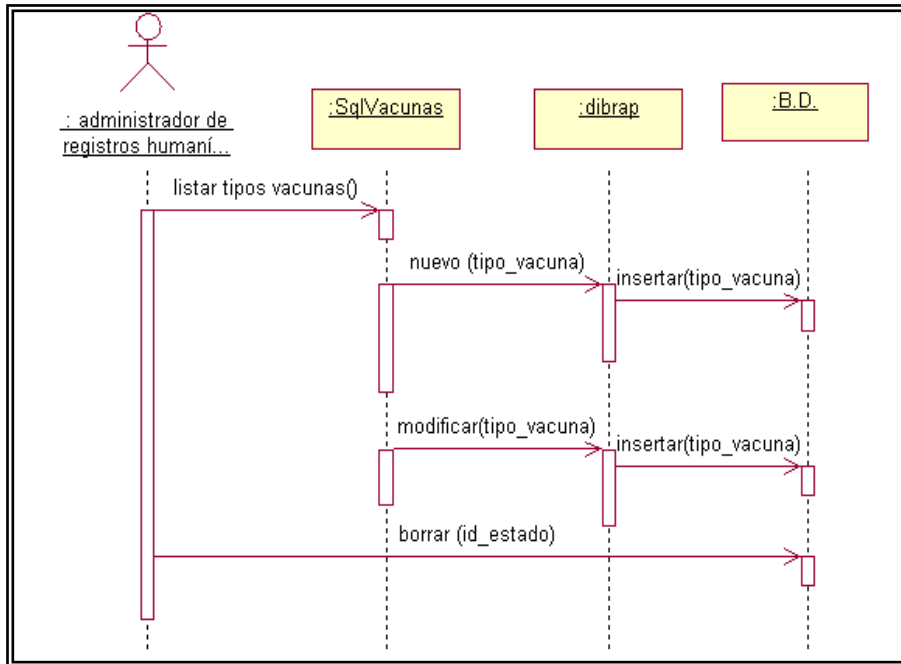


Fig. 3.6.5.23: Diagrama de secuencia registrar tipo de vacunas
Fuente. Elaboración propia

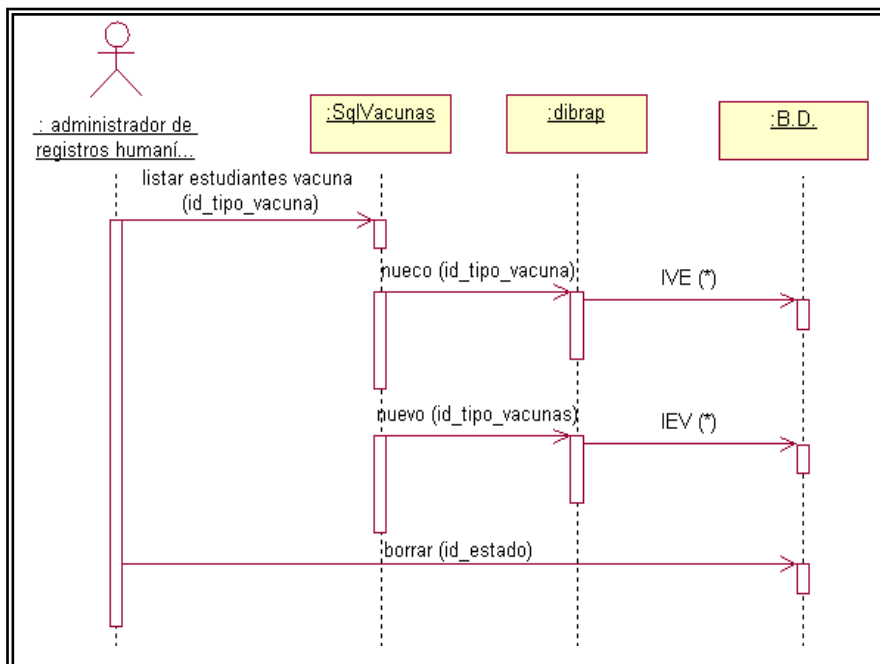


Fig. 3.6.5.24: Diagrama de secuencia registrar tipo de vacunas de estudiantes
Fuente. Elaboración propia

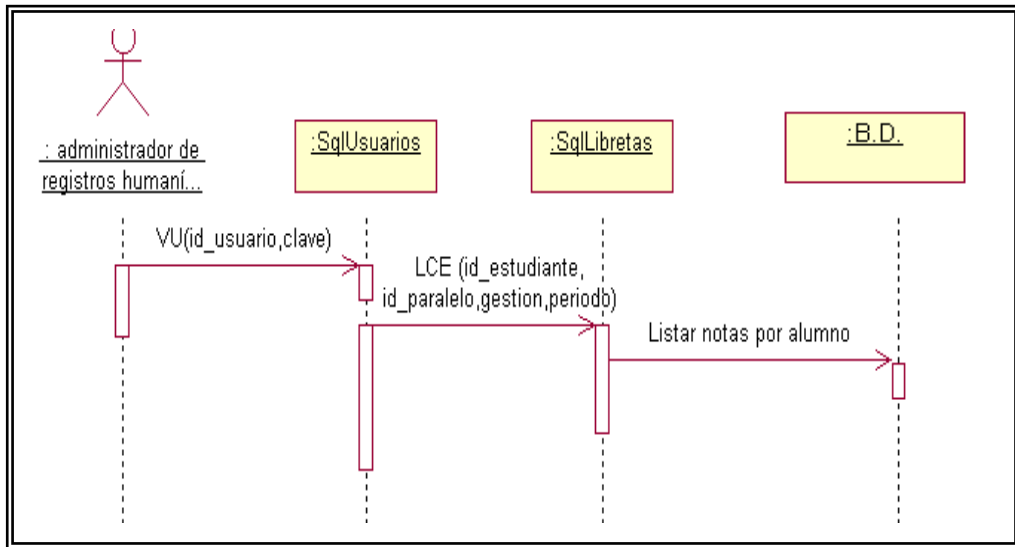


Fig. 3.6.5.25: Diagrama de secuencia Listar notas por alumno
Fuente. Elaboración propia

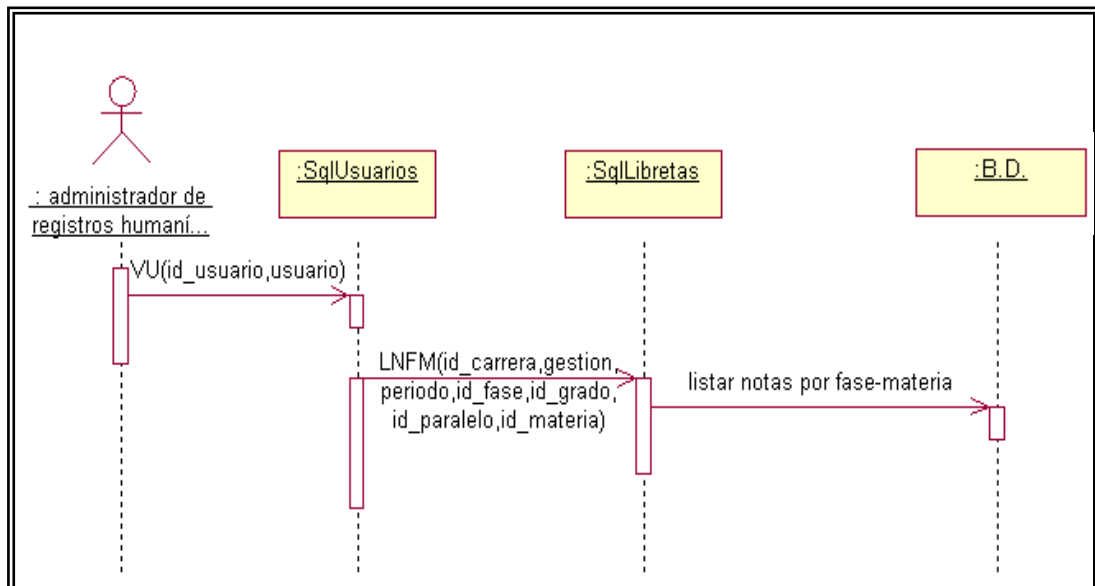


Fig. 3.6.5.26: Diagrama de secuencia listar notas por fase - materia
Fuente. Elaboración propia

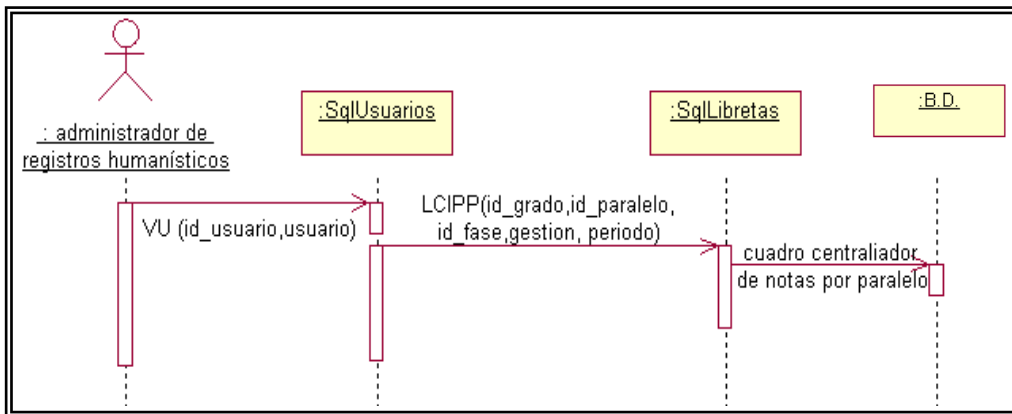


Fig. 3.6.5.27: Diagrama de secuencia listar notas por fase - libretas
Fuente. Elaboración propia

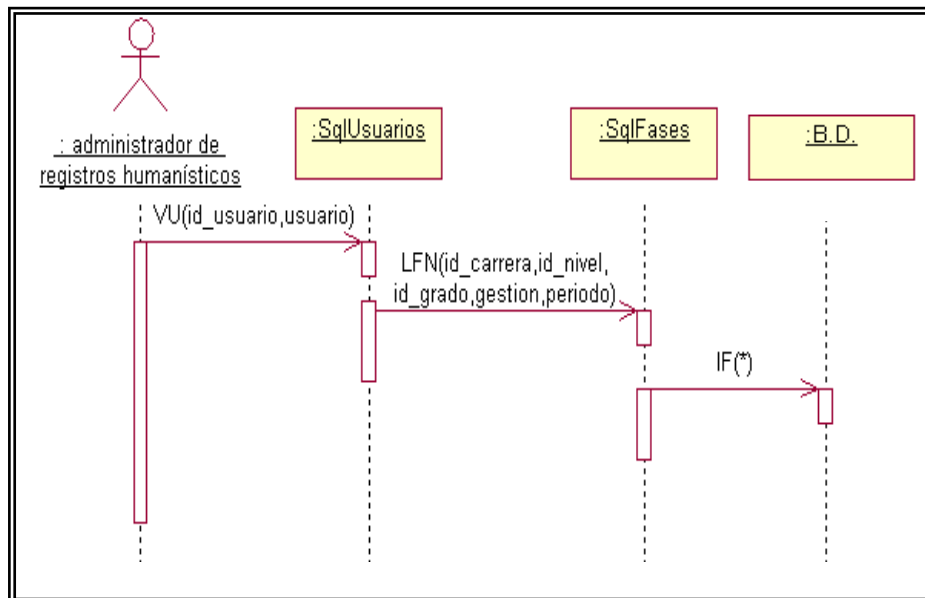


Fig. 3.6.5.28: Diagrama de secuencia abrir fase
Fuente. Elaboración propia

IF(*):id_fase,id_carrera,id_nivel,gestion,periodo,fase,ponderacion,nota maxima,nota_aprobacion,descripcion,id_rol

3.6.6. Diagrama de secuencia desde el punto de vista del Estudiante

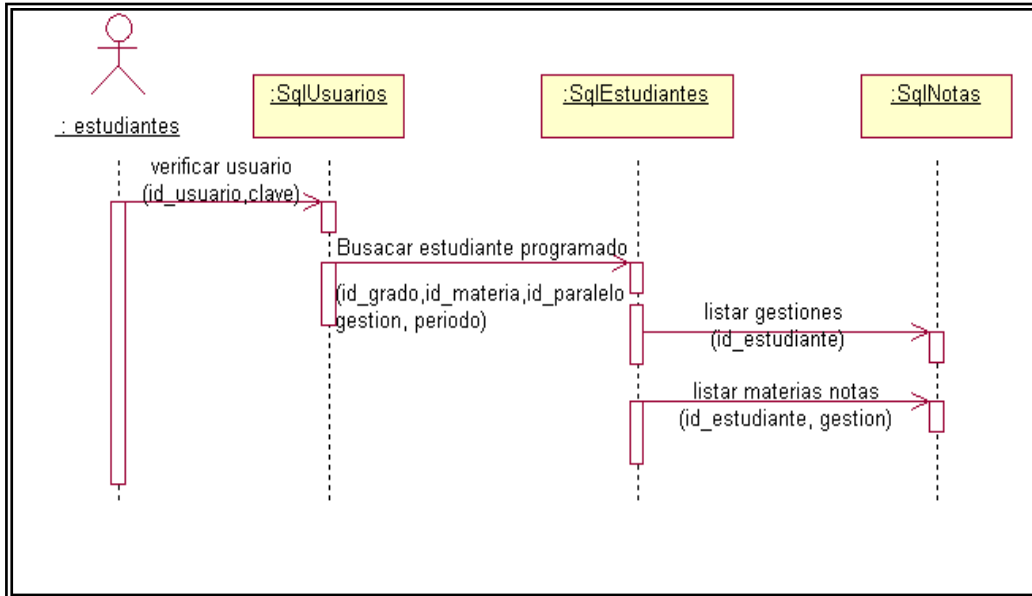


Fig. 3.6.6.1: Diagrama de secuencia ver historial académico
Fuente. Elaboración propia

3.6.7. Diagrama de secuencia desde el punto de vista del Docente

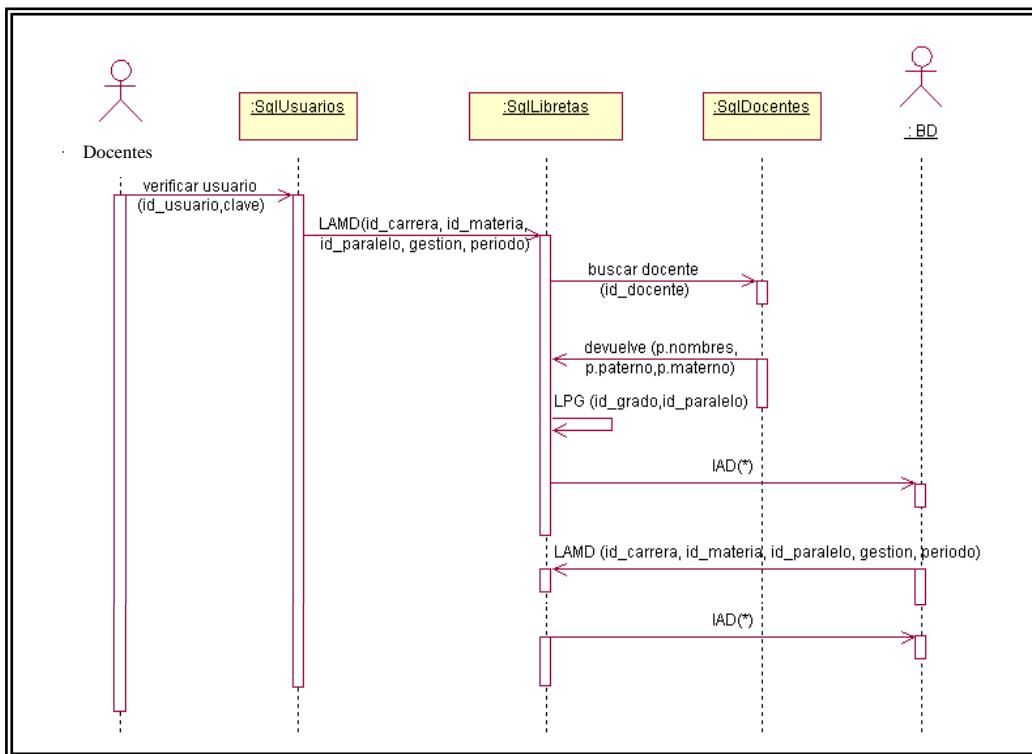
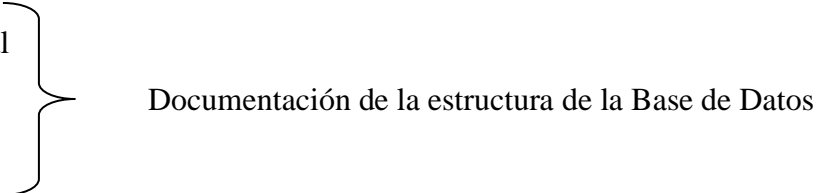


Fig. 3.6.7.1: Diagrama de secuencia Administrar libretas
Fuente. Elaboración propia

3.7. INGENIERIA INVERSA DE BASE DE DATOS

El primer paso fue, identificar las tablas que conforman el sistema de Información Académico-Administrativo, para la adecuación del sistema en la unidad educativa Bruno Racua, este sistema tiene una base de datos relacional y cuenta con 157 tablas.

Debido a la inexistencia de documentación de la base de datos del sistema implementado, es que se realiza el diseño de la base de datos del sistema información, utilizando método de ingeniería inversa en base de datos, de las cuales se realizaron las siguientes etapas:

- Diseño conceptual
 - diseño lógico
 - diseño físico
- 
- Documentación de la estructura de la Base de Datos

Previo desarrollo a las etapas del ingeniería inversa, se describe de forma general las tablas involucradas en el Sistema de Información Académico-Administrativo, asimismo se especifica la cantidad de tablas utilizadas en el proyecto.

3.7.1. DOCUMENTACIÓN DE LA ESTRUCTURAR DE BASE DE DATOS

El diseño físico describe datos que son almacenados en la base de datos y las relaciones que existen entre los mismos, este diseño es descrito por el sistema manejador de base de datos (esquema interno).

a) Diseño Conceptual de la base de datos

Las bases de datos son una parte fundamental, para cualquier sistema informático y administración de la misma, se elabora el diccionario de la base de datos con la finalidad de facilitar el manejo y control de la base de datos, ofreciendo la guía para administradores y programadores. El diccionario de datos completo se encuentra en el anexo A

Diccionario de Datos

Nombre de la tabla: est_libretas						
Detalle de la Tabla						
Campos	Tipo	Primaria	Foráneas	Valor x Defecto	Dependencia_Tabla	Descripción
id_libreta	serial	si		no nulo		auto numérico
id_estudiante	dentero		si		estudiantes	
id_grado	dentero		si		grados	
id_paralelo	dentero		si		paralelos	
id_materia	dentero		si		materias	
id_fase	dentero		si		fases	
id_sub_fase	dentero2		si	0	sub_fases	
id_estado	did_estado		si	A	_estados	X es anulado, A es activo
id_tipo_nota	dentero		si		tipo_notas	
gestion	dentero					
periodo	dentero					
nro_nota	dentero					
nota	dnota2			0		
fec_registro	dfec_modificacion					
fec_modificacion	dfec_modificacion					
id_rol	int4			0	roles	
ult_usuario	did_usuario				usuarios	

Tabla 3.7.1: Estudiantes Libretas

Fuente: Elaboración propia

Nombre de la tabla: docentes						
Detalle de la Tabla						
Campos	Tipo	Primaria	Foraneas	Valor x Defecto	Dependencia_Tabla	Descripción
id_docente	serial	si		no nulo		auto numérico
id_persona	dentero		si		personas	
id_estado	did_estado		si	'A',X	_estados	X es anulado, A es activo
apodo	dtexto					
clave	dtexto2			DEFAULT ('123456':::dtexto2)::dtexto2		
fec_registro	dfec_modificacion					
fec_modificacion	dfec_modificacion					
ult_usuario	did_usuario				usuarios	

Tabla 3.7.2: Docentes

Fuente: Elaboración propia

Nombre de la tabla: aulas						
Detalle de la Tabla						
Campos	Tipo	Primaria	Foráneas	Valor x Defecto	Dependencia_Tabla	Descripción
id_aula	serial	si		no nulo		auto numérico
id_tipo_aula	dentero		si		tipos_aulas	
id_estado	did_estado		si	'A'	_estados	X es anulado, A es activo
aula	dtexto2					
capacidad	dentero2					
superficie	dentero2					
piso	dentero2					
bloque	dtexto2					
fec_registro	dfec_modificacion					
fec_modificacion	dfec_modificacion					
ult_usuario	did_usuario				usuarios	

Tabla 3.7.3: Aulas
Fuente: Elaboración propia

Nombre de la tabla: estudiantes						
Detalle de la Tabla						
Campos	Tipo	Primaria	Foráneas	Valor x Defecto	Dependencia_Tabla	Descripción
id_estudiante	serial	si		no nulo		auto numérico
id_persona	dentero		si		personas	
id_carrera	dentero		si		carreras	
id_plan	dentero		si		planes	
id_tipo_admision	dentero		si		tipo_adminisiones	
id_tipo_estudiante	dentero				tipo_estudiantes	
id_estado	did_estado		si	'A'	_estados	X es anulado, A es activo
fec_inscripcion	dfecha					
fec_egreso	dfecha2					
fec_registro	dfec_modificacion					
fec_modificacion	dfec_modificacion					
id_rol	int4			0	roles	
ult_usuario	did_usuario				usuarios	

Tabla 3.7.4: Estudiantes
Fuente: Elaboración propia

b) Diseño lógico de la base de datos

En esta etapa primero se realiza una extracción del estructura existente en el sistema informático, este etapa se realiza mediante un comando del DBMS PostgreSQL, el mismo que también permite obtener todo al estructura de la base de datos como fueron diseñados inicialmente.

Pg_dump nombre_base_datos > nombre_base_datos.sql

Nombre de la base de datos existentes

Nombres de la copia de base de datos

Al ejecutar este comando se obtiene el diseño físico de la base de datos, a continuación se publican algunas de las tablas:

```
CREATE TABLE _categorias (  
  id_categoria serial NOT NULL,  
  id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
  categoria dtexto,  
  orden dentero2 DEFAULT 1,  
  imagen dtexto2 DEFAULT ('categoria.gif'::dtexto2)::dtexto2,  
  fec_registro dfec_modificacion,  
  fec_modificacion dfec_modificacion,  
  ult_usuario did_usuario  
);
```

```
CREATE TABLE personas (  
  id_persona serial NOT NULL,  
  id_pais dentero,  
  id_departamento dentero,  
  id_provincia dentero,  
  id_seccion dentero,  
  id_canton dentero,  
  id_localidad dentero,  
  id_zona dentero,  
  id_tipo_estado_civil dentero,  
  id_tiposexo dentero,  
  id_tipo_grupo_sanguineo dentero,  
  id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
  nombres dtexto,  
  paterno dtexto2,  
  materno dtexto2,  
  dip dtexto2,  
  fec_nacimiento dfecha2,  
  direccion dtexto2,  
  telefono dtexto2,  
  celular dtexto2,  
  correo dtexto2,  
  anio_titulacion dentero2,  
  titulo dtexto2,  
  nro_hijos dentero2,  
  nro_dependientes dentero2,  
  nro_seguro_medico dtexto2,  
  fec_registro dfec_modificacion,  
  fec_modificacion dfec_modificacion,  
  ult_usuario did_usuario  
);
```

```
CREATE TABLE estudiantes (  
  id_estudiante serial NOT NULL,  
  id_persona dentero,  
  id_carrera dentero,  
  id_plan dtexto,  
  id_tipo_admision dentero,  
  id_tipo_estudiante dentero,  
  id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
  fec_inscripcion dfecha,  
  fec_egreso dfecha2,  
  fec_registro dfec_modificacion,  
  fec_modificacion dfec_modificacion,  
  id_rol integer DEFAULT 0,  
  ult_usuario did_usuario );
```

```
CREATE TABLE grd_paralelos (  
    id_grd_paralelo serial NOT NULL,  
    id_carrera dentero,  
    id_materia dentero4,  
    id_grado dentero,  
    id_paralelo dentero,  
    id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
    gestion dentero,  
    periodo dentero,  
    cupo_actual dentero2 DEFAULT 0,  
    cupo_max dentero2 DEFAULT 80,  
    fec_registro dfec_modificacion,  
    fec_modificacion dfec_modificacion,  
    id_rol integer DEFAULT 0,  
    ult_usuario did_usuario );
```

```
CREATE TABLE docentes (  
    id_docente serial NOT NULL,  
    id_persona dentero,  
    id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
    apodo dtexto,  
    clave dtexto2 DEFAULT ('123456'::dtexto2)::dtexto2,  
    fec_registro dfec_modificacion,  
    fec_modificacion dfec_modificacion,  
    ult_usuario did_usuario  
);
```

```
CREATE TABLE lbr_tipos_notas (  
    id_tipo_nota serial NOT NULL,  
    id_fase dentero,  
    id_sub_fase dentero2 DEFAULT 0,  
    id_carrera dentero,  
    id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
    gestion dentero,  
    periodo dentero,  
    tipo_nota dtexto,  
    fec_registro dfec_modificacion,  
    fec_modificacion dfec_modificacion,  
    id_rol integer DEFAULT 0,  
    ult_usuario did_usuario );
```

```
CREATE TABLE notas (  
  id_nota serial NOT NULL,  
  id_matricula dentero,  
  id_estudiante dentero,  
  id_materia dentero,  
  id_paralelo dentero,  
  id_convalidacion dentero2 DEFAULT 0,  
  id_estado did_estado DEFAULT ('A'::did_estado)::did_estado,  
  gestion dentero,  
  periodo dentero,  
  nota dentero2 DEFAULT 0,  
  observacion dtexto2,  
  fec_registro dfec_modificacion,  
  fec_modificacion dfec_modificacion,  
  id_rol integer DEFAULT 0,  
  ult_usuario did_usuario );
```

c) **Diseño Físico**

En esta parte del diseño físico de la base de datos se realizara una descripción de las estructuras de almacenamiento.

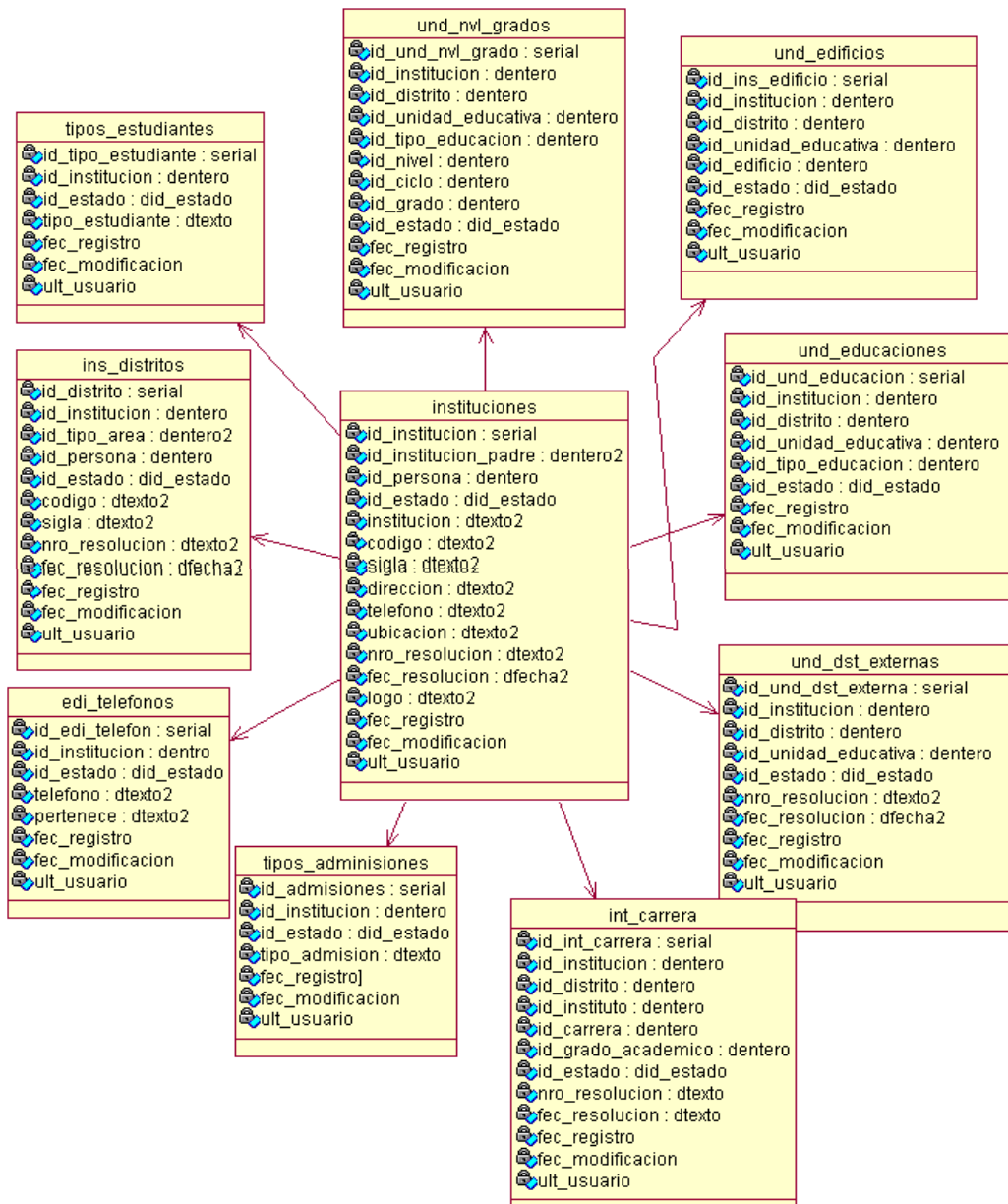


Figura 3.7.5: Base de Datos

Fuente: Elaboración propia

3.8. INTERFAZ DEL SISTEMA

La interfaz del sistema es el medio por el cuál se establece un diálogo entre el sistema y usuario, el medio de comunicación puede ser física mediante el teclado o simbólico por medio de iconos.

El sistema de información académico administrativo consta de una interfaz de usuario para la introducción de datos (entorno Web), en esta interfaz el usuario realiza el llenado de la base de datos con todos los atributos que se le da a cada usuario.

3.8.1. INTERFAZ DE USUARIO PARA LA INTRODUCCIÓN DE DATOS

Permite el ingreso al sistema, especificando su cuenta de usuario hizo clave (PIN) del mismo.

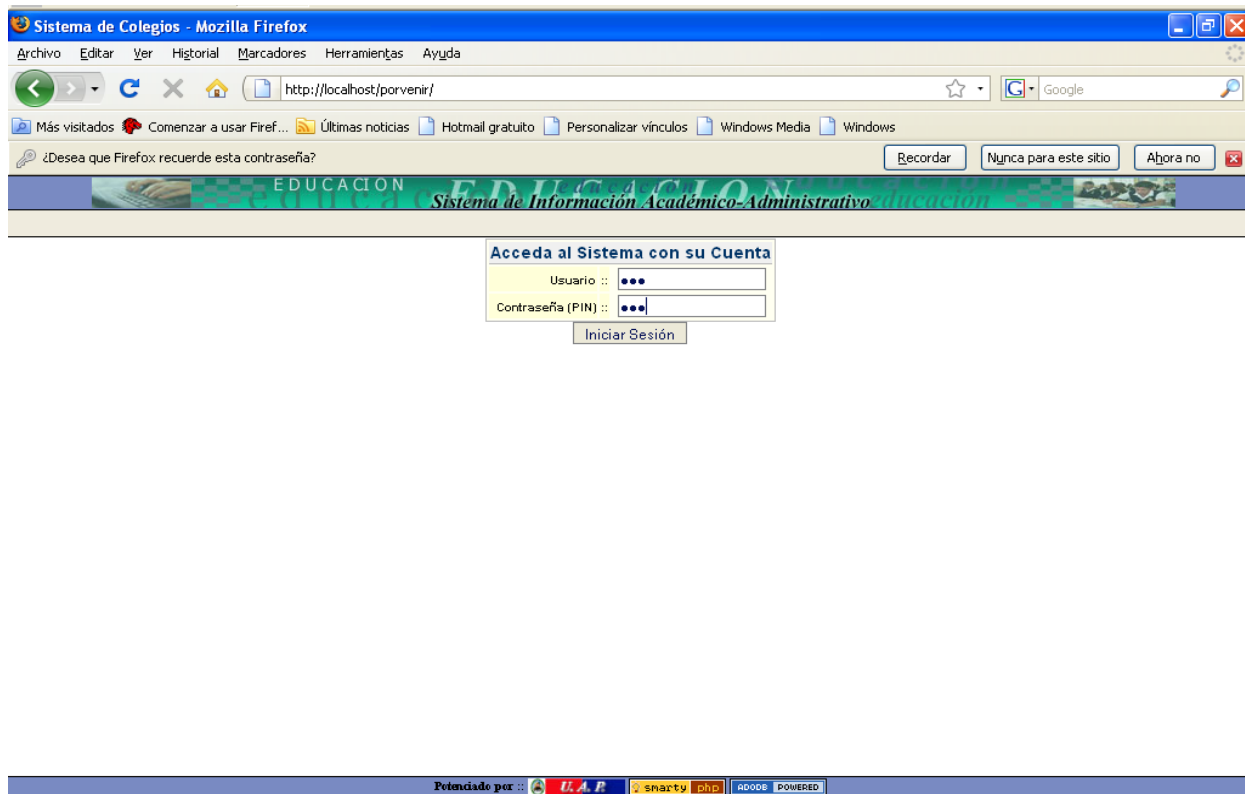


Figura 3.8.1: Interfaces de usuario- validación de usuario
Fuente: Sistema de Información Académica-Administrativo

Figura3.8.2: El usuario administrador de registros humanísticos accede al sistema, a continuación se le ofrecen las distintas opciones, que se muestran a continuación.

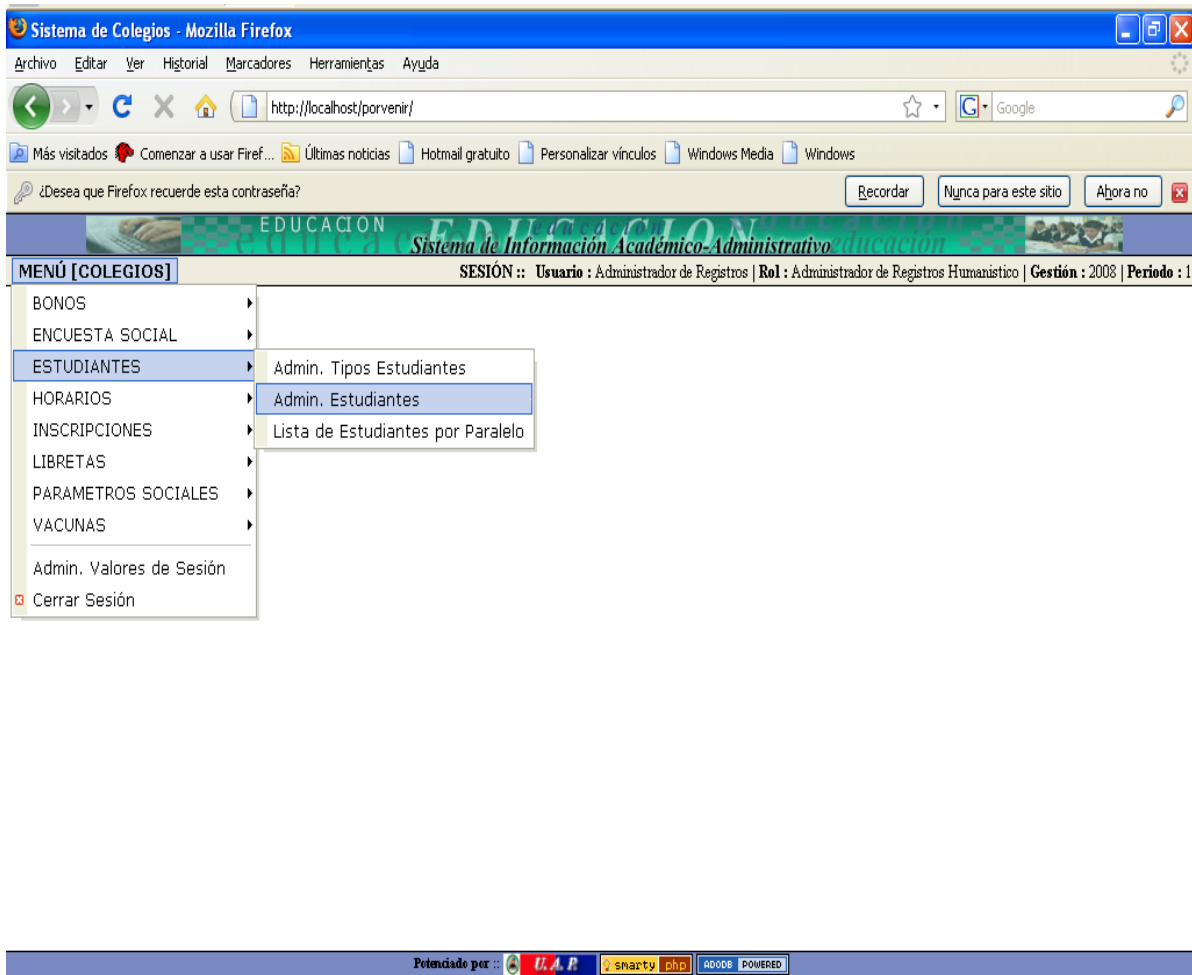


Figura 3.8.2: Interfaces de usuario- opciones del usuario
Fuente: Sistema de Información Académica-Administrativo

Figura 3.8.3: Al elegir una de las acciones desplegadas del menu podrá realizar la manipulación de los datos de acuerdo a lo seleccionado.

Sistema de Colegios - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://localhost/porvenir/

EDUCACION *Sistema de Información Académico-Administrativo*

MENÚ [COLEGIOS] SESIÓN :: Usuario : Administrador de Registros | Rol : Administrador de Registros Humanistico | Gestión : 2008 | Período : 1

Admin. Estudiantes

Filtro :: contiene Filtrar Ver todo

1 - 50 de 164
Páginas: 1 2 3 4

Seleccionar: Todos, Ninguno, Invertir

<input type="checkbox"/>	Borrar	Nº	Id_persona	Carrera	Plan	Tipo Admisión	Tipo Estudiante	Fecha Inscripción	Fecha Egreso	id_rol	Modificar
<input type="checkbox"/>		1	Estudiante	Bachillerato	1	Nuevo	Nuevo	2008-01-24	1000-01-01	0	Modificar
<input type="checkbox"/>		2	"ALEJANDRA" "LLANOS" "CARDENAS"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		3	"ALEJANDRO MICHEL" "TICONA" "BAUTISTA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		4	"ALEX ALEJANDRO" "VASQUEZ" "CAMARGO"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		5	"ANNIE" "AGUADA" "TIRINA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		6	"ARIEL" "MACUAPA" "CUADIAY"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		7	"CARLOS" "HAENSELL" "SIRIGUA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		8	"CAROLINA" "PINEDO" "SOLARES"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		9	"CRISTIAN JESUS" "VACA" "SOLIZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-19	1000-01-01	0	Modificar
<input type="checkbox"/>		10	"DANIEL" "CONDORI" "BALBOA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		11	"DANITZA" "CORTEZ" "MIRANDA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		12	"DAYANA" "PEREZ" "DA SILVA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		13	"DELM" "TORREZ" "ALVAREZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-21	1000-01-01	0	Modificar
<input type="checkbox"/>		14	"DENIS" "MELENA" "LOPEZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		15	"DEVORA" "HURTADO" "GONGORA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		16	"DIANA" "RODRIGUEZ" "VASQUEZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-30	1000-01-01	0	Modificar
<input type="checkbox"/>		17	"DRIANA" "MEJIDO" "MANIGUARY"	Bachillerato	1	Antiguo	Antiguo	2008-01-20	1000-01-01	0	Modificar
<input type="checkbox"/>		18	"ELENITA" "SIVORA" "CHAO"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		19	"ELIER" "ALVAREZ" "CHUQUI"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		20	"EVA" "ALBUQUEIQUE" "VITURINO"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		21	"GRESLY" "JUSTINIANO" "DURAN"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar
<input type="checkbox"/>		22	"GUADALUPE" "AGUADA" "MELGAR"	Bachillerato	1	Antiguo	Antiguo	2008-01-29	1000-01-01	0	Modificar
<input type="checkbox"/>		23	"GUSTAVO" "SIVORA" "CHAO"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modificar

Potenciado por: U.A.R. Smarty PHP 80008 POWERED

Figura 3.8.3: Interfaces de usuario- Administrar Estudiantes
Fuente: Sistema de Información Académica-Administrativo

Figura 3.8.4: Esta figura permite el ingreso al la manipulación y administración de inscripciones de los estudiantes

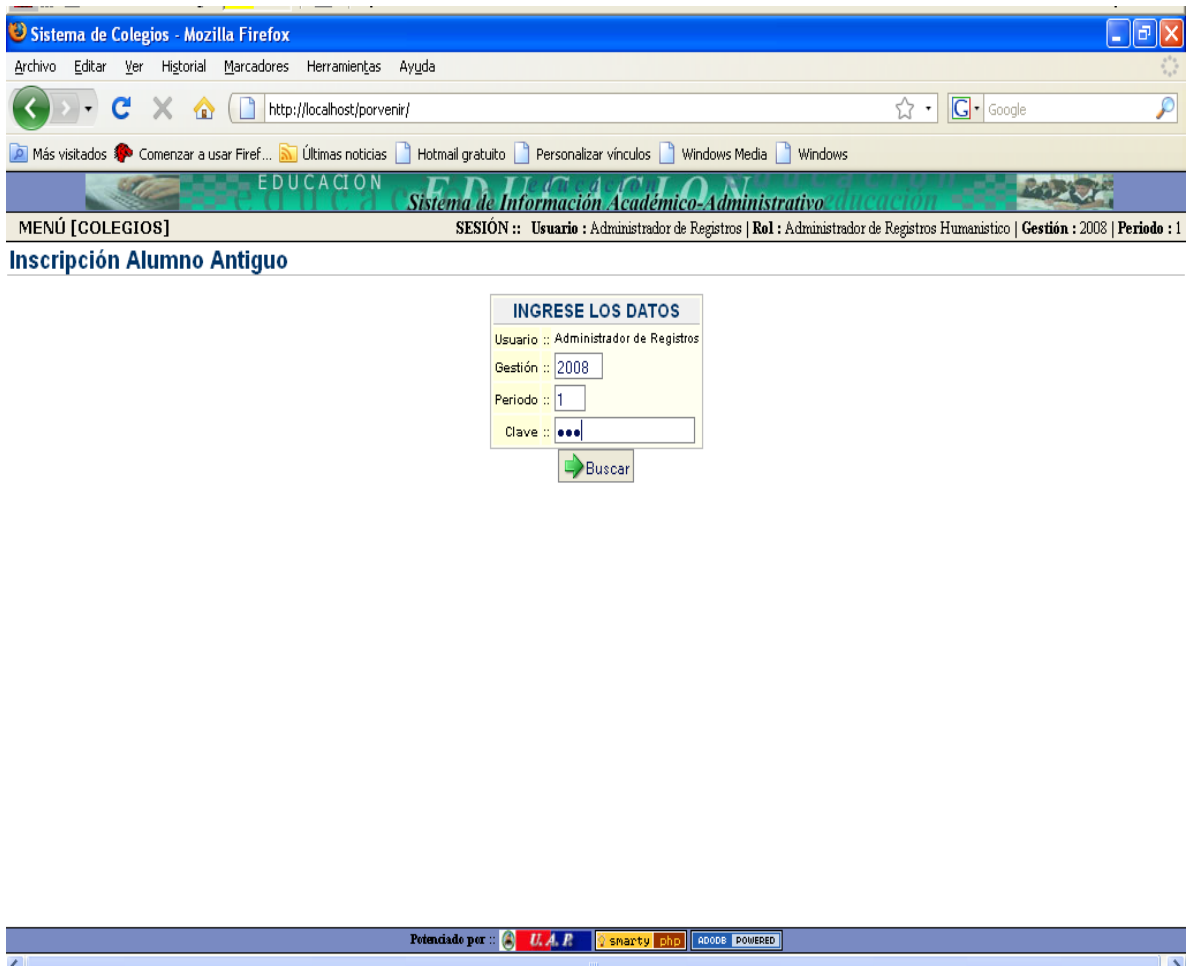


Figura 3.8.4: Interfaces de usuario- inscripción alumnos antiguos
Fuente: Sistema de Información Académica-Administrativo

Figura 3.8.5: Esta figura muestra los datos que deben ser designados de forma obligatoria con el fin de poder administrar las inscripciones de los estudiantes.

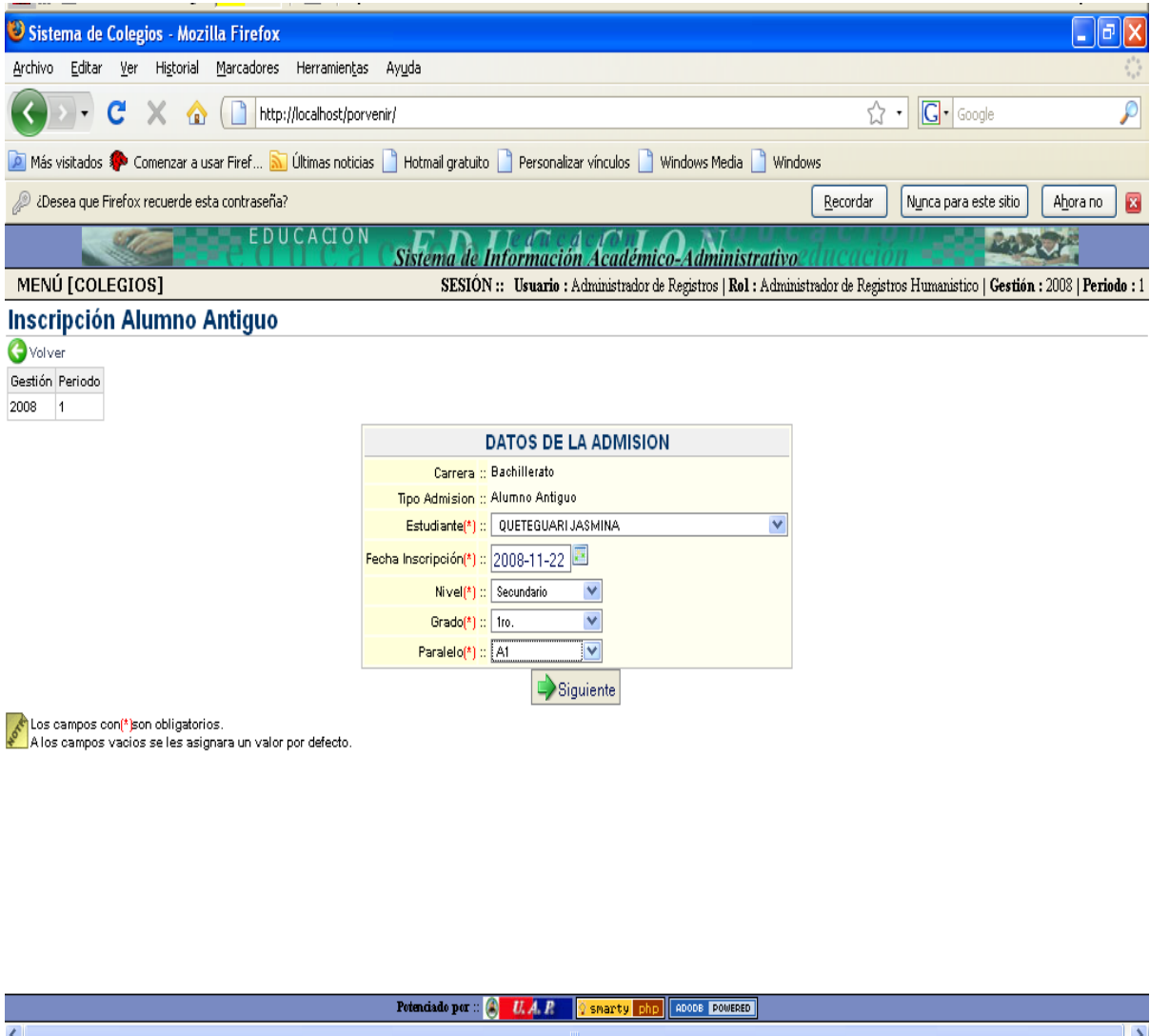


Figura 3.8.5: Interfaces de usuario- inscripción alumnos antiguos (datos de la admisión)
Fuente: Sistema de Información Académica-Administrativo

Figura 3.8.6: Esta pantalla permite realizar búsquedas de personas, carrera, plan, tipo de admisión, tipo estudiante, fecha de inscripción, fecha de egreso, etc., dichas búsquedas se les puede realizar de 2 formas: la primera por medio del *filtro* donde se elige el tipo de búsqueda que se desea, la segunda búsquedas se realiza apretando *ctrl.+F*, donde se describe el nombre de la búsqueda.

The screenshot shows a web browser window titled 'Sistema de Colegios - Mozilla Firefox'. The address bar shows 'http://localhost/porvenir/'. The page header includes 'EDUCACION Sistema de Información Académico-Administrativo'. Below the header is a navigation menu with 'Admin. Estudiantes' selected. A search filter is visible with 'Tipo Estudiante' selected in a dropdown menu and 'contiene' in another. Below the filter is a table of student records with columns: 'Borrar', 'Nº', 'Id_persona', 'Carrera', 'Plan', 'Tipo Admisión', 'Tipo Estudiante', 'Fecha Inscripción', 'Fecha Egreso', 'id_rol', and 'Modific.'. The table contains 15 rows of student data. At the bottom of the page, there is a search bar with the text 'Encontrar:' and a search button. An arrow points to the search bar with the label 'Segunda forma'. Another arrow points to the filter dropdown menu with the label 'Primera forma'.

Borrar	Nº	Id_persona	Carrera	Plan	Tipo Admisión	Tipo Estudiante	Fecha Inscripción	Fecha Egreso	id_rol	Modific.
<input type="checkbox"/>	1	Estudiante	Bachillerato	1	Nuevo	Nuevo	2008-01-24	1000-01-01	0	Modif
<input type="checkbox"/>	2	"ALEJANDRA" "LLANOS" "CARDENAS"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	3	"ALEJANDRO MICHEL" "TICONA" "BAUTISTA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	4	"ALEX ALEJANDRO" "VASQUEZ" "CAMARGO"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	5	"ANNIE" "AGUADA" "TIRINA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	6	"ARIEL" "MACUAPA" "CUADIAY"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	7	"CARLOS" "HAENSELL" "SIRIGUA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	8	"CAROLINA" "PINEDO" "SOLARES"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	9	"CRISTIAN JESUS" "VACA" "SOLIZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-19	1000-01-01	0	Modif
<input type="checkbox"/>	10	"DANIEL" "CONDORI" "BALBOA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	11	"DANITZA" "CORTEZ" "MIRANDA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	12	"DAYANA" "PEREZ" "DA SILVA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	13	"DELMÍ" "TORREZ" "ALVAREZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-21	1000-01-01	0	Modif
<input type="checkbox"/>	14	"DENIS" "MELENA" "LOPEZ"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif
<input type="checkbox"/>	15	"DEVORA" "HURTADO" "GONGORA"	Bachillerato	1	Antiguo	Antiguo	2008-01-11	1000-01-01	0	Modif

Figura 3.8.6: Interfaces de usuario- búsqueda

Fuente: Sistema de Información Académica-Administrativo

Figura 3.8.7: Un ejemplo de reporte de la lista de estudiantes por grado y su paralelo correspondiente.

COLEGIO NACIONAL BRUNO RACUA
 Porvenir Fono: 842
 Pando-Bolivia

LISTA DE ALUMNOS

GRADO : 3ro. PARALELO : A3 GESTION : 2008

Nro.	Identificador	Paterno	Materno	Nombres
1	48	ARANA	AGUILERA	YENNY
2	49	ARANA	AGUILERA	YENNY
3	50	BISMARCK	GUARENA	MARIELY
4	51	CARBALLO	TIRINA	FERNANDO
5	54	CATY	NOSA	IDILIO
6	75	DE LIMA	FLORES	JORGE
7	55	FERNANDEZ	BECERRA	SEBASTIAN
8	53	FERNANDEZ	HAENSEL	ELIEXY
9	57	FERNANDEZ	HAENSEL	ERWIN
10	58	GUARY	VALLANUEVA	DANNYS
11	59	GUERRA	RAMIREZ	EVARISTO
12	61	GUTIERREZ	HAENSELL	LUCIA ELENA
13	60	GUZMAN	NAY	GELBER
14	64	HERBERA	HERBERA	HERBERA

Encontrar: Siguierte Anterior Resaltar todo Coincidencia de mayúsculas/minúsculas

Terminado

Figura 3.8.7: Reporte lista de estudiantes

Fuente: Sistema de Información Académica-Administrativo

El presente capítulo describe la implementación y las pruebas realizadas al Sistema de Información Académico-Administrativo en la Unidad Educativa Bruno Racua, mediante el diagrama de despliegue, como también se especifican las configuraciones realizadas para el funcionamiento del sistema, a demás se utilizo las técnicas de pruebas de caja negra (pruebas unitarias).

5.1 IMPLEMENTACION

Una vez concluida la etapa de ingeniería inversa de procesos e ingeniera de base de datos, se tiene una visión clara de los procesos que realiza el sistema, de tal manera se procede a la implementación del Sistema de Información Académico-Administrativo en la unidad educativa Bruno Racua en la localidad de Porvenir, para posteriormente realizar las pruebas del mismo.

Siguiendo con la metodología descrita en anteriores capítulos a continuación se presenta el Diagrama de despliegue, para mostrar los elementos lógicos, sus ubicaciones físicas y como se comunican estos elementos.

5.2 DIAGRAMA DE DESPLIEGUE

PROPUESTO

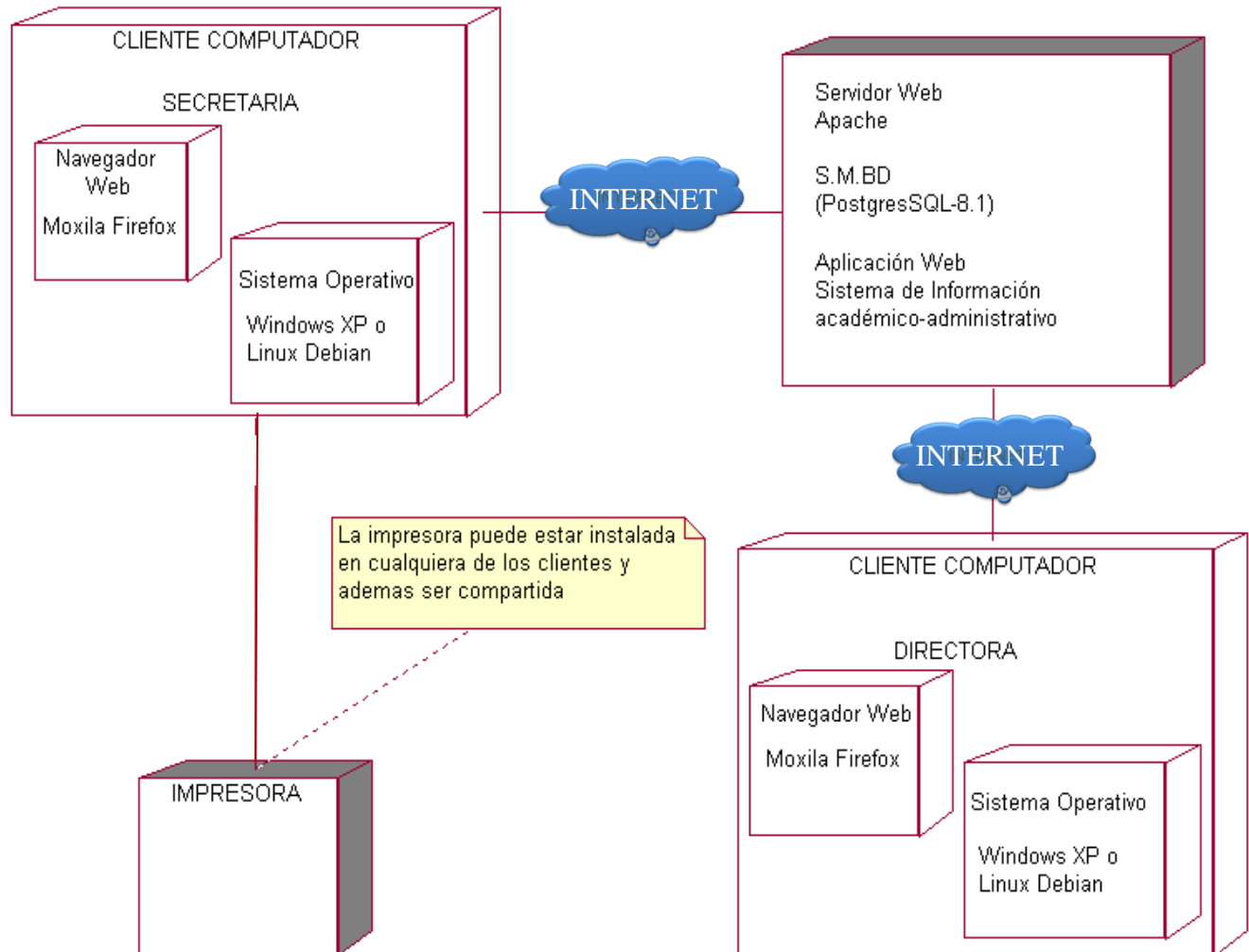


Figura5.1: Diagrama de despliegue propuesto
Elaboración: Propia

Al principio del proyecto se considero que el Sistema de Información trabajaría con un servidor y dos maquinas que trabajarían como clientes, también se considero que el sistema trabajaría con internet, pero por fallas de material para realizar la conexión de una intranet y por fallas con el proveedor de servicio de Internet, no se pudo concretar. Sin embargo el sistema fue desarrollado tomando en cuenta estos imprevistos, y una vez solucionados estos desafíos el sistema funcionara con toda normalidad.

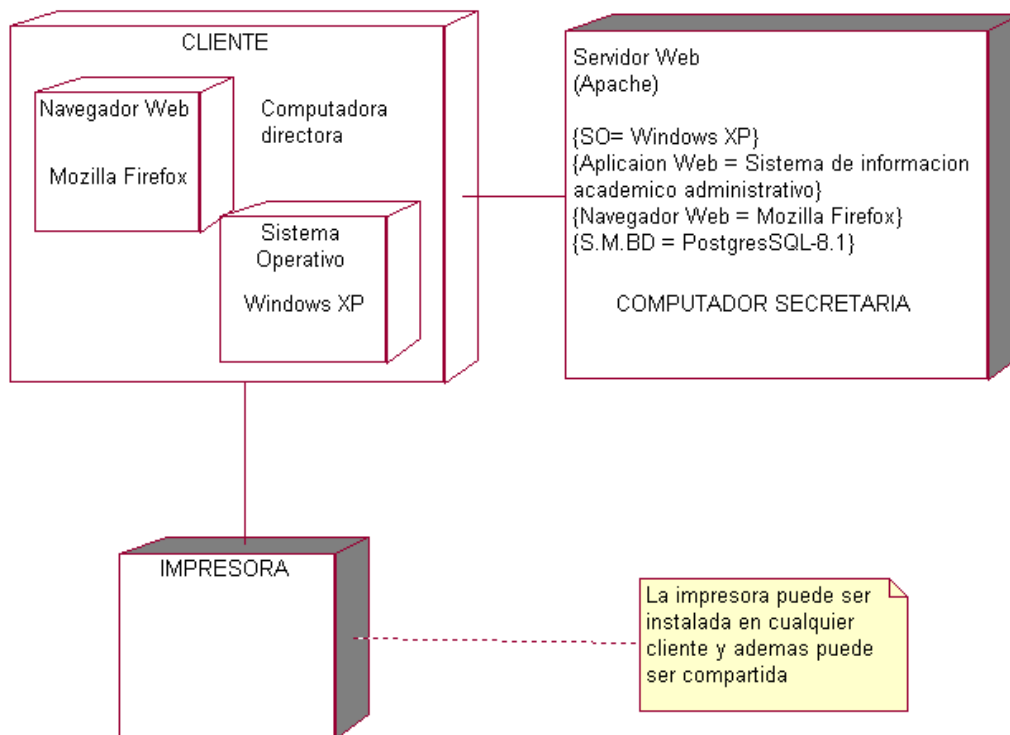
MODIFICADO

Figura5.2: Diagrama de despliegue sistema modificado
Elaboración: Propia

5.3 ACTIVIDADES DE LA IMPLEMENTACION

Cabe mencionar que a raíz de la vista de implementación estática del sistema por medio del diagrama de despliegue anteriormente ilustrado, a continuación se detallan las actividades llevadas a cabo para la implementación del Sistema de Información Académico-Administrativo.

➤ **Instaladores**

- Se debe realizar la copia de los instaladores en la unidad C del equipo. Este conjunto de instaladores está compuesto por: PEAR, postgresql-8.1, appserv-win32-2.5.9, Firefox Setup 3.0., estos son herramientas de trabajo para el buen funcionamiento del sistema.

➤ **Plataforma del Sistema Operativo**

La plataforma utilizada en el Presente Proyecto es la de Windows XP, el sistema es multi plataforma (funciona bajo diferentes sistemas operativos), por lo tanto el sistema funciona también bajo el Sistema Operativo de Linux en su distribución Debian 8.1zz

➤ **Postgresql 8.1**

Se realiza la instalación del sistema manejador de Base de Datos, teniendo en cuenta la creación de las tablas en el formato UTF8, esto para que las tablas reconozcan algunas datos que contengan letras o signos, como ser la ñ o Ñ.

➤ **Configuración de Variables de PostgreSQL 8.1**

En el menú principal ubicar PostgreSQL 8.1, y realizar la modificación en los siguientes archivos.

- Configuration files
- Edtpg-hba.conf

Cambiar mds a trust

➤ **Creación de nuevo rol**

Dentro del PgAdminIII (este es una aplicación grafica para gestionar el gestor de la base de datos de PostgreSQL), se creara el nuevo rol, los pasos a seguir son los siguientes.

- Roles de login (clic derecho)
 - Nuevo rol de login
 - Se habilitan toda las opciones
 - Nombre que se le dará al nuevo rol USER

➤ **Instalación y configuración del Appserv**

- Se descomprime el archivo y se pasa a la instalación
- Las tablas tienen que estar en el formato UTF8
- Las carpetas adod y smarty se copian en la siguiente dirección: C/Appser /php5/PPEAR
- Dentro de C/Appser /php5, se encuentran los siguientes archivos:
- php.init.recomend, una vez dentro del archivo eliminar el ; de la extensión php_pda_pgsql
- En el mismo archivo eliminar el ; de la extensión include_path y además direccionar el archivo con la dirección del PEAR

➤ **Sistema**

- El sistema de Información académico-administrativo se debe copiar en la siguiente dirección: C/ Appser/www/Sistema de colegios.
- Copiar la Base de Datos “BDcol” a la siguiente dirección: C/Archivos de Programa/ Postgresql/8.1/bin

➤ **Migrar los datos**

Con el siguiente comando se migrarán los nuevos datos a la Base de Datos

Pg_dump nombre_base_datos > nombre_base_datos.sql

Nombre de la base de datos existentes

Nombres de la copia de base de datos

Una vez terminada la implementación se pasó a la capacitación del personal responsable de utilizar el sistema, Prof. Mery Oshiro (Directora de la unidad educativa Bruno Racua) y a la Sra. Karin Aguada (Secretaria del establecimiento).

5.4. PRUEBAS

Las pruebas que se utilizaron para la verificación de los resultados de la implementación del sistema, son: la prueba unitaria de caja negra y la norma de calidad ISO 9126. Dado a la magnitud del sistema se tomo en cuenta los subsistemas de más relevancia.

5.4.1. PRUEBAS UNITARIAS

- **Método de Partición Equivalente**

1.- C.I. número de cedula de identidad que identifica de manera única a una persona.

Condiciones de Entrada	Clases de equivalencia Validas	Clases de equivalencias
Tipo de dato	1.- Entero Positivo	2.- Otro tipo
Cantidad de dígitos	3.- = 7	4.- $\diamond 7$
El primer digito empieza	5.- >0	6.- = 0

Tabla 5.4.1: Caso de Prueba CI
Fuente: Elaboración Propia

CASOS DE PRUEBA

N°	Condiciones de Entrada	Clases de equivalencia Validas	Cubre las Clases
1	1767266	Muestra datos encontrados	1,3,5

Tabla5.4.2: Caso de Prueba CI N1
Fuente: Elaboración Propia

N°	Condiciones de Entrada	Resultados esperados	Cubre las Clases invalidadas
1	-1234567	Muestra mensaje de no existir coincidencia	2

2	123	Muestra mensaje de no existir coincidencia	4
3	000234566	Muestra mensaje de no existir coincidencia	4
4	Gwuei2	Muestra mensaje de no existir coincidencia	6

Tabla5.4.3: Caso de Prueba CI N2
Fuente: Elaboración Propia

2.- Apellido Paterno mas apellido Materno mas Nombre, identifican a una persona

Condiciones de Entrada	Clases de equivalencia Validas	Clases de equivalencias
Tipo de dato	1.- Alfabeto	2.- Otro tipo
Cantidad de caracteres	3.- <= 30	4.- >30
Cantidad de Palabras	5.- >=1 ,6.- >=3	7.- >3

Tabla5.4.4: Caso de Prueba Ap. Paterno
Fuente: Elaboración Propia

CASOS DE PRUEBA

N°	Condiciones de Entrada	Resultados esperados	Cubre las Clases
1	Oshiro	Muestra apellido paterno	1,3,5
2	Duran	Muestra apellido materno	1,3,5,6

Tabla 5.4.5: Caso de Prueba Ap. Paterno N1
Fuente: Elaboración Propia

N°	Condiciones de Entrada	Resultados esperados	Cubre las Clases invalidadas
1	Gwuei2	Muestra mensaje de no existir coincidencia	2
2	Hh,#r3	Muestra mensaje de no	4

		existir coincidencia	
--	--	----------------------	--

Tabla 5.4.6: Caso de Prueba Ap. Paterno N2
Fuente: Elaboración Propia

3.- Apellido Paterno identifican a una persona

N°	Condiciones de Entrada	Resultados esperados	Cubre las Clases
1	Oshiro	Muestra apellido paterno	1,3,5
2	Duran	Muestra apellido materno	1,3,5,6

Tabla 5.4.7: Caso de Prueba Paterno
Fuente: Elaboración Propia

N°	Condiciones de Entrada	Resultados esperados	Cubre las Clases
1	*&OsHiro	Muestra mensaje de no existir coincidencia	2
2	OShiro Duran	Muestra mensaje de no existir coincidencia	4
3	122Oshiro	Muestra mensaje de no existir coincidencia	7

Tabla 5.4.8: Caso de Prueba Paterno N1
Fuente: Elaboración Propia

4.- Nombre identifica el nombre completo de una persona

Condiciones de Entrada	Resultados esperados	Cubre las Clases
Mery	Muestra Nombre	1,3,5
Mery Eva	Muestra Nombre	1,3,5,6

Tabla 5.4.9: Caso de Prueba Paterno N2
Fuente: Elaboración Propia

Condiciones de Entrada	Resultados esperados	Cubre las clases invalidadas
##Mer	Muestra mensaje de no existir coincidencia	1,3,5
.....))Mery	Muestra mensaje de no existir coincidencia	1,3,5,6

Tabla 5.4.10: Caso de Prueba Paterno N3
Fuente: Elaboración Propia

5.- Usuario verifica el nombre de usuario y su clave

Condiciones de Entrada	Clases de equivalencia Validas	Clases de equivalencias
Tipo de dato	1.- Texto	2.- Otro tipo
Cantidad de caracteres	3.-30	4.- >10
Cantidad de Palabras	5.- >=1 ,6.- >=60	7.- >3

Tabla 5.4.11: Caso de Prueba Usuario
Fuente: Elaboración Propia

Condiciones de Entrada	Resultados esperados	Cubre las Clases
Regente	Verifica usuario	1,3,5
adr	Verifica Clave	1,3,5

Tabla 5.4.12: Caso de Prueba Usuario N1
Fuente: Elaboración Propia

Condiciones de Entrada	Resultados esperados	Cubre las clases invalidadas
#@Regente	Muestra mensaje no puede encontrar usuario	2,3,5
ADR	Muestra mensaje de no existir coincidencia	1,3,5
Re\$%genTEweq	Muestra mensaje de no existir	2,4

	coincidencia	
A.....	Muestra mensaje de no existir coincidencia	2,4

Tabla 5.4.13: Caso de Prueba Usuario N2
Fuente: Elaboración Propia

5.4.2. Calidad de Software:

De acuerdo a la norma ISO/IEC 9126 se ha elaborado un cuestionario de evaluación (**Ver Anexo B**), mediante el cual se puede medir la calidad del software tomando esto los resultados obtenidos. El sistema fue evaluado por el usuario con más acceso a todo el sistema.

El cuestionario de evaluación tiene las siguientes características:

- El peso fue asignado en base al grado de importancia de cada uno de los atributos en el software.
- La escala de calificaciones para el software va desde 1 que significa muy poco importante, hasta 10 que significa extremadamente importante.
- Las preguntas de la evaluación se formularon de acuerdo al atributo exigido.

Los resultados del cuestionario de evaluación permitieron aplicar la formula para obtener las métricas de cada atributo.

$$\text{Medida de Sub característica} = \frac{R_1M_1 + R_2M_2 + \dots + R_mM_m}{R_1 + R_2 + \dots + R_m}$$

Se califica un puntaje desde 1 hasta 5, de acuerdo a la siguiente escala:

CALIFICACIÓN	PUNTAJE
Muy mala	1
Mala	2
Medida	3
Buena	4
Muy buena	5

TABLA 5.4.2. Escala de calificación del cuestionario de evaluación.

FUENTE: Elaboración propia.

- **Funcionalidad**

Atributo	Peso	Resultado
Adecuación	10	4
Seguridad	10	4

TABLA 5.4.3 Resultados obtenidos en las características de funcionalidad

FUENTE: *Elaboración propia*

$$\text{Funcionalidad} = \frac{10 * 4 + 10 * 4}{10 + 10} \quad ; \quad \text{Funcionalidad} = 4$$

El resultado obtenido se aproxima al valor máximo, lo que significa que el sistema cumple con uno de los requisitos más importantes exigidos en la etapa inicial del desarrollo del proyecto.

- **Fiabilidad**

Atributo	Peso	Resultado
Madurez	8	3
Recuperabilidad	8	3
Conformidad de la fiabilidad	10	5

TABLA 5.4.4 Resultados obtenidos en la característica de fiabilidad.

FUENTE: *Elaboración propia.*

$$\text{Fiabilidad} = \frac{8 * 3 + 8 * 3 + 10 * 5}{10 + 8 + 8} \quad ; \quad \text{Fiabilidad} = 3,7$$

El resultado de fiabilidad es de 3,7; lo que indica que cumple con lo solicitado por el usuario.

- **Usabilidad**

Atributo	Peso	Resultado
Entendimiento	10	5
Aprendizaje	8	3
Operabilidad	8	4
Atracción	6	4

TABLA 5.4.5 Resultados obtenidos en las características de usabilidad.

FUENTE: Elaboración propia.

$$\text{Usabilidad} = \frac{10 * 5 + 8 * 3 + 8 * 4 + 6 * 4}{10 + 8 + 8 + 6} ; \quad \text{Usabilidad} = 4$$

Una vez evaluadas las sub características de *usabilidad* se puede evidenciar que cuenta con buena facilidad de entendimiento y aprendizaje.

- **Eficiencia**

Atributo	Peso	Resultado
Utilización de recursos	10	4,5
Comportamiento de tiempos	10	4

TABLA 5.4.6 Resultados obtenidos en la característica de eficiencia

FUENTE: Elaboración propia.

$$\text{Eficiencia} = \frac{10 * 4,5 + 10 * 4}{10 + 10} ; \quad \text{Eficiencia} = 4,2$$

El sistema ha mejorado en gran manera el proceso de orientación vocacional, lo que a permitido obtener la información con mayor rapidez.

- **Mantenimiento**

Atributo	Peso	Resultado
Capacidad de ser analizado	9	4
Cambiabilidad	10	5
Estabilidad	10	5
Facilidad de prueba	10	5

TABLA 5.4.7 Resultados obtenidos en la característica de mantenimiento.
FUENTE: Elaboración propia.

$$\text{Mantenimiento} = \frac{9 * 4 + 10 * 5 + 10 * 5 + 10 * 5}{9 + 10 + 10 + 10} ; \quad \text{Mantenimiento}=4,7$$

El puntaje obtenido es de 4,7, lo que indica que el sistema cuenta con la estabilidad y mantenibilidad necesaria.

- **Portabilidad**

Atributo	Peso	Resultado
Adaptabilidad	9	4
Facilidad de instalación	7	4
Coexistencia	8	4

TABLA 5.48 Resultados obtenidos en las características de portabilidad.
FUENTE: Elaboración propia.

$$\text{Portabilidad} = \frac{9 * 4 + 7 * 4 + 8 * 4}{9 + 7 + 8} ; \quad \text{Portabilidad} = 4$$

El resultado obtenido de 4, lo que nos indica que el sistema puede coexistir con otros sistemas.

Los resultados que de forma individual se obtuvieron aun no muestran el grado de calidad respecto a las métricas, establecidas por el atributo. En lo que sigue se determina la calidad la calidad del Sistema completo, donde los atributos ahora consideran factores en la medición de la calidad.

R1=10	B1=4	Funcionalidad
R2=10	B2=3.7	Confiabilidad
R3=10	B3=4	Usabilidad
R4=10	B4=4.2	Eficiencia
R5=8	B5=4	Portabilidad
R6=9	B6=4.7	Facilidad de Mantenimiento

$$\text{Factor} = \frac{10*4 + 10*3.7 + 10*4 + 10*4.2 + 8*4 + 10*4.7}{10+10+10+10+10+8} = \frac{238}{58} = \mathbf{4.1}$$

Tomando en cuenta el criterio de los atributos de funcionalidad, fiabilidad, usabilidad y eficiencia son aspectos importantes en la medición de la calidad del software, y de acuerdo a la evaluación realizada al sistema de información Académico- Administrativo, se obtuvo un resultado de **4.1**, es decir un valor adecuado y bueno

El presente capítulo describe las conclusiones generales del Trabajo Dirigido, en lo que se refiere a la descripción de logros alcanzados, de la misma forma se describen las recomendaciones para el buen funcionamiento del sistema

5.1. CONCLUSIONES

Con la Implementación del Sistema de Información Académico-Administrativo, en los predios de la Unidad Educativa Bruno Racua, en la localidad de Porvenir y la ejecución del Trabajo Dirigido, se puede concluir lo siguiente.

- Representación del funcionamiento estático y dinámico, del Sistema de Información Académico-Administrativo, detallado mediante los casos de Diagramas de Uso y Diagramas de Secuencia.
- Se tiene analizada y documentada la Base de Datos mediante el diseño conceptual, diseño lógico y diseño físico.
- Normalización de la información obtenida de las hojas de cálculo, en relación a la información de la Unidad Educativa Bruno Racua, y migración de los mismos a la base de datos del Sistema de Información Académico - Administrativo.
- Sistema de Información Académico – Administrativo implementado, contribuyendo a la mejora de la gestión académica, en la Unidad Educativa Bruno Racua.
- Usuarios capacitados y aptos en el manejo del Sistema de Información Académico – Administrativo.

5.2. RECOMENDACIONES

Con la conclusión del presente trabajo dirigido se pone en consideración las siguientes recomendaciones

- Se recomienda a la Unidad Educativa realizar convenios con SEDUCA, para tomar en cuenta a un proveedor de internet, de tal manera que el sistema pueda funcionar bajo una plataforma Web, para una mayor facilidad en el manejo.
- El Sistema de Información Académico-Administrativo fue desarrollado bajo la Ley 1565 de Reforma Educativa, y cualquier cambio en esta ley debe ser considerado en el sistema, para futuras modificaciones del mismo.
- Para un mejor acceso y facilidad de los estudiantes al Sistema de Información Académico-Administrativo, se propone implementar el uso de tarjetas magnéticas, para que de esta manera se elimine el uso de contraseñas
- Considerar el desarrollo del modulo de Seguridad, sacando copias de seguridad de la base de datos en tiempos determinados, de forma automatizada, para evitar pérdidas de información por diversos factores.