



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**



FACULTAD DE INGENIERÍA

**SISTEMA EN LÍNEA PARA LA EVALUACIÓN
DOCENTE EN UNA INSTITUCIÓN EDUCATIVA**

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTAN

FUENTES HERNÁNDEZ JULIO CÉSAR
GALEANA PIEDRA SELENE NALLELY
MARTINEZ JACINTO OMAR ANTONIO
ZEPEDA MADRIGAL DANIEL JESÚS

DIRECTOR DE TESIS : M.I. JUAN CARLOS ROA BEIZA

Ciudad Universitaria, México D.F. 2013



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE

CAPITULO I. ENTORNO DEL PROBLEMA

1.1 INTRODUCCIÓN	2
1.2 CONCEPTOS BÁSICOS DE LA EVALUACIÓN DOCENTE	5
1.3 SITUACIÓN ACTUAL DE LA EVALUACIÓN DOCENTE EN LA INSTITUCIÓN EDUCATIVA.	11
1.4 ORGANIGRAMA DE LA INSTITUCIÓN Y RESPONSABILIDADES DE CADA ÁREA	14
1.5 METODOLOGÍAS RECOMENDADAS PEDAGÓGICAMENTE	20

CAPITULO II. MARCO TEÓRICO

2.1 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE LA METODOLOGÍA RUP Y LENGUAJE UML.	32
2.2 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DEL LENGUAJE DE PROGRAMACIÓN PHP Y HTML	51
2.3 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DEL LENGUAJE DE PROGRAMACIÓN JAVASCRIPT	57
2.4 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DEL LENGUAJE DEL MODELO DE BASES DE DATOS RELACIONALES	61
2.5 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DEL MOTOR DE BASES DE DATOS RELACIONALES. POSTGRESQL	71

CAPITULO III. ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA

3.1 ANÁLISIS DEL PROBLEMA.	76
3.2 RECOPIACIÓN Y ANÁLISIS DE LA INFORMACIÓN	80
3.3 REQUERIMIENTOS GENERALES Y ESPECÍFICOS DE LA APLICACIÓN	96
3.4 PLANTEAMIENTO DE LA SOLUCIÓN Y POSIBLES MÓDULOS	102
3.5 JUSTIFICACIÓN DE LA METODOLOGÍA Y SOFTWARE A UTILIZAR	108

CAPITULO IV. DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN	
4.1 MODELO DEL SISTEMA	122
4.1.1 DIAGRAMA DE CASOS DE USO	122
4.1.2 DIAGRAMA DE MODELADO DE DATOS	125
4.1.3 DIAGRAMA ENTIDAD-RELACIÓN (DER)	126
4.1.4 DIAGRAMA DE CLASE	130
4.1.5 DICCIONARIO DE DATOS	133
4.2 IMPLEMENTACIÓN DE LA BASE DE DATOS	144
4.3 DISEÑO DE INTERFAZ DE USUARIO	171
4.4 GENERACIÓN DE PRUEBAS Y MANTENIMIENTO	179
4.5 GENERACIÓN DE REPORTE	201
CONCLUSIONES	204
BIBLIOGRAFÍA	206

CAPITULO I. ENTORNO DEL PROBLEMA

CAPÍTULO I. ENTORNO DEL PROBLEMA

1.1 Introducción

Entre las múltiples acciones que se pueden realizar para el mejoramiento de la calidad de la educación, la evaluación del profesor juega un papel muy importante, pues permite recopilar información sobre su desempeño y, por lo tanto, orientar las acciones encaminadas a propiciar estrategias de desarrollo profesional.

Alcanzar la anhelada calidad de la enseñanza requiere profesores 100% comprometidos y adecuadamente formados en sus respectivas materias, capaces de planificar, desarrollar y transmitir los conocimientos que ellos tienen hacia los alumnos.

Para incentivar la mejora de estos, las universidades elaboran un sistema de evaluación a los profesores basados en la opinión de los alumnos.

Existen tres principales razones por las que esta fuente de información se selecciona como la vía preferente en la evaluación del profesor:

- Permite que el alumnado opine sobre sus docentes
- Se considera que los estudiantes, por su naturaleza, son los observadores más extensos de la enseñanza, y a su vez están en una posición única para valorar la calidad del curso, la dedicación y la preparación del profesor.
- La confiabilidad de las observaciones realizadas por el alumno es normalmente elevada. Ésta depende, entre otros factores, del número de encuestados, por lo que si se seleccionan las muestras adecuadas la confiabilidad puede estar asegurada.

En la actualidad, dichas evaluaciones al profesor se llevan a cabo en papel, un comité que toma la responsabilidad de auditor aplica dichas evaluaciones a todos los grupos de la universidad, teniendo que organizarse en diferentes días para abarcar a todos los alumnos. Se presentan al salón con las evaluaciones ocasionando que la clase sea interrumpida, las reparten a los alumnos y una vez contestado en su totalidad estas son entregadas para luego colocarlas en un sobre el cual será sellado para después contabilizar las respuestas. Este procedimiento debe de hacerle salón por salón.

Una vez que recaban las evaluaciones de todos los alumnos, proceden a contabilizar las respuestas pero al no hacer uso en su totalidad de un sistema de cómputo, los diversos indicadores de retroalimentación para los profesores pueden demorar mucho tiempo, además de que no se asegura totalmente su confiabilidad, ya que las evaluaciones pueden presentar tachaduras o correcciones en las diversas respuestas por lo que estas no se tomarían en cuenta.

Por esta razón es que elegimos este tema ya que como ingenieros podemos aportar nuestro conocimiento para desarrollar un sistema el cual innove el proceso de evaluación al profesor, para que sea lo más automático posible y reduzca el tiempo y el número de personas que intervienen en el proceso, además de asegurar la confiabilidad de los indicadores y una mayor generación de reportes en menor tiempo, los cuales serán evaluados por un comité pedagógico para que realmente las preguntas tengan relevancia.

Los indicadores ayudaran a orientar al profesor para saber sus áreas de oportunidad, recibir capacitación y cursos.

El sistema hará a un lado el uso de papel para las evaluaciones y la necesidad de que un comité tenga que presentarse en cada salón para aplicar la evaluación. Esta se llevara a cabo en línea por lo que el alumno la podrá llevar a cabo si así lo

prefiere fuera del salón de clases, evitando así la influencia del profesor que algunas veces es factor para que el alumno responda de cierto modo, en la hora y lugar que él decida contando con computadora y acceso a internet.

El sistema se tiene contemplado desarrollar con herramientas gratuitas, actuales y eficientes, con la capacidad suficiente para procesar la información recabada.

De tal manera que nuestra tesis estará agrupada en 4 capítulos.

En el primer capítulo abarcaremos la situación actual de la evaluación al profesor, explicaremos los conceptos básicos con los cuales se evalúa además de las metodologías las cuales se utilizan para la selección de las preguntas que estarán contenidas en la evaluación.

Dentro del segundo capítulo explicaremos el por qué la utilización del lenguaje UML, PHP, HTML, Java Script y PostgreSQL, mostrando las características, ventajas y desventajas de cada una de estas metodologías o lenguajes.

En el capítulo tres de la presente tesis hacemos referencia al análisis del problema, con el cual una vez entendido, plantearemos la posible solución a través de una aplicación.

En el último capítulo mostramos como se lleva a cabo la implantación de esta aplicación entrando a los detalles como la base de datos la interfaz al usuario y el resultado final reflejado en los reportes que servirán para retroalimentación hacia los profesores.

Y por último daremos nuestras conclusiones sobre las diferencias y logros obtenidos de hacer el proceso de evaluación de manera manual a llevarlo a cabo de una forma automática apoyándonos con una aplicación específica para esto.

1.2 Conceptos básicos de la Evaluación Docente

La evaluación de la docencia es un proceso mediante el cual a través de la aplicación de un cuestionario a un conjunto de alumnos en una entidad universitaria, éstos evalúan las características y habilidades de sus profesores durante la clase.

Asegurar la calidad de la docencia es el objetivo de la evaluación. Se trata de un proceso sistemático diseñado para medir de forma objetiva el rendimiento laboral del docente. Para que tenga operatividad es preciso que se trate de un proceso ó modelos con las siguientes características:

Sistemático: La evaluación debe ser una actividad que debe hacerse periódicamente, en función de un programa previamente consensuado. La evaluación debe tener un propósito y ese propósito debe estar en consonancia con el modelo pedagógico.

- **Objetivo:** Se debe garantizar, en la medida de lo posible la objetividad de las evaluaciones, a lo que contribuirá la existencia de diversas fuentes de información.
- **Participativo:** Toda la organización, empezando por el propio evaluado, debe participar en la difusión de la evaluación.
- **Flexible:** Se trata de un sistema, no de una técnica. Ello significa que debe elegirse el método de evaluación en función de las características de la organización.
- **Utilidad:** se debe asegurar que la evaluación proporcione información útil y oportuna a los docentes, de esta manera, los informes deben ser claros, exactos y oportunos. En este sentido, la evaluación debe tener una

orientación constructiva que permita apoyar a los docentes en su desarrollo profesional.

- **Viabilidad:** se debe asegurar que la evaluación sea realista, prudente y, en medida de lo posible, fácil de aplicar, haciendo uso de procedimientos prácticos que produzcan la información necesaria del objeto de evaluación.
- **Exactitud:** se debe garantizar que la evaluación revele y transmita información técnica precisa sobre las características que determinan el valor o mérito del objeto de evaluación. Así mismo, los procedimientos de obtención de información deben ser fiables y asegurar que las conclusiones se encuentren lógicamente vinculadas con los datos obtenidos.
- **Propiedad (legalidad y ético):** se debe asegurar que la evaluación se conduzca de manera legal y ética con pleno respeto al bienestar profesional de los evaluados.

De acuerdo con la enciclopedia general de la educación (1999), se ha definido la evaluación, fundamentalmente como un proceso con fases características. Por lo tanto una evaluación docente implica una metodología mediante la cual se recopila información sobre la práctica docente a través de medios o instrumentos con el fin de juzgar su valor y tomar decisiones. Las etapas se pueden clasificar en tres:

- La planificación y diseño de la evaluación involucra actividades de establecimiento de un propósito, finalidad y función, delimitar los procedimientos de la evaluación (modelo e instrumentación), definir los agentes que efectuarán la evaluación y establecer la temporalización de la acción evaluativa.

- El desarrollo, recogida y tratamiento de la información implica las actividades de obtención de la información, codificación, registro, análisis , es decir tratamiento de la información.
- La evaluación y utilización de los resultados evaluativos comprende el análisis de resultados y la formulación de juicios (interpretación, clasificación y comparación), toma de decisiones, divulgación de los resultados, negociación con las audiencias y seguimiento.

Con el propósito de guiar el desarrollo del modelo, se enuncian nueve principios retomados de diversos autores y enfoques que se consideran necesarios para el diseño y puesta en marcha de dicho modelo, a fin de orientar sus acciones:

- Orientación formativa. El modelo permitirá al profesor reflexionar y retroalimentarse sobre su acción en la docencia, y plantear acciones para la mejora de la misma (Zabalza, 1990).
- Orientación participativa. La evaluación/formación en el contexto de la práctica docente y educativa, será elaborada e instrumentada en conjunto por autoridades y docentes de la institución, así el docente se involucrará y participará en su diseño.
- Orientación humanista. Considerará al docente como una persona, un ser humano, con preocupaciones, intereses, necesidades, expectativas y emociones, de ahí que buscará la preservación de su dignidad, autoestima e individualidad (Loredo y Rigo, 2001).
- Enfoque multidimensional. Tomará en consideración los rasgos, acciones, conductas y actitudes en torno a la relación cotidiana maestro–alumno en el aula (Barbier, 1999; Doyle, 1986).

- Enfoque multirreferencial. Convocará a una pluralidad de puntos de vista, de perspectivas aplicadas sobre un objeto, sin por ello intentar ejercer un control total de este objeto; además advertirá sobre la complejidad de la práctica educativa docente y con ello se acercará a muchos de los factores, hechos y situaciones que la determinan (Ardoino, 2005; Barbier, 1999).
- Enfoque ecológico. Reconocerá la complejidad de la vida en las aulas y la convertirá en su principal objeto de estudio, en particular lo referente al trabajo articulado y conjunto que realizan profesores y estudiantes para alcanzar los objetivos educativos. Estudiará la naturaleza de los procesos sociales desde la perspectiva de los participantes, por lo que atenderá el carácter contextual de los procesos investigados (Shulman, 1989).
- Evaluación/Formación contextualizada. Considerará que la práctica del docente no es neutra, ni está aislada, por ello deberá incluirse en su aplicación el marco normativo y las condiciones institucionales en que opera. La institución educativa en la que se lleva a cabo la práctica educativa docente, tiene una historia, lineamientos y formas de trabajo que influyen sobre el ejercicio docente, por ello la institución deberá plantear de forma clara cuál es su misión y su visión, su modelo curricular, su modelo pedagógico, su perfil ideal del docente, sus políticas y sus metas (Loredo, 2000).
- Considerar la acción docente como una práctica reflexiva. La reflexión en la acción para el docente universitario se posicionará como un diálogo continuo que implica la construcción de una teoría sobre el caso en cuestión, indagará las descripciones adecuadas en torno a una situación, definirá de manera interactiva los medios y los fines, y reconstruirá y reevaluará las prácticas que se llevan a cabo (Schön, 1992).

- Considerar al docente como un agente activo. Considerará la habilidad que tiene el profesor para solucionar problemas prácticos y operar situaciones complejas. Aprovechará los resultados de la evaluación de su trabajo, para reflexionar críticamente a propósito de sus fortalezas y debilidades (Schön, 1992).

Tomando en cuenta los principios antes mencionados, consideramos que el modelo de evaluación del personal docente tendrá que estar, cuando menos, relacionado con los siguientes aspectos:

- Evaluación de la actividad instructiva (aula y tutorías).
 - Opinión de los alumnos.
- Evaluación de la actividad investigadora.
 - Propone actividades extra clase para reforzar lo aprendido en clase.
 - Propone temas actuales y discuten sobre ellos.
- Evaluación de la actividad departamental.
 - Asiste puntualmente.
 - Prepara su clase.
- Evaluación de las condiciones de trabajo del profesor.

Es importante tomar en cuenta las condiciones tecnológicas y recursos didácticos con los que cuenta la institución en las que se desarrolla el trabajo del profesor evaluado.

Se ha realizado un análisis factorial con objeto de identificar los factores que puedan ser utilizados para representar las relaciones entre las variables consideradas, habiendo obtenido tres factores. Estos factores han recibido las siguientes denominaciones:

- Factor I. Formación ética y profesional

Evalúa la frecuencia con la que el profesor enseña a respetar las opiniones de los compañeros de clase, trata a los alumnos de manera equitativa, los respeta, es ético, permite la participación de los alumnos y el diálogo abierto, evalúa el aprendizaje de manera objetiva, enseña los valores éticos de la profesión y domina los temas.

- Factor II. Estrategias de enseñanza

Valora con qué frecuencia el profesor asiste puntualmente a impartir clase, exhorta a los alumnos a respetar el horario de la clase, prepara su clase y la imparte de manera clara, vincula la teoría con la práctica, enseña a los alumnos a comprender los temas, realiza evaluaciones sólo de los temas revisados en clase, revisa los trabajos que deja, resuelve dudas y hace recomendaciones para que los alumnos tengan un buen desempeño en la carrera y en la vida profesional.

- Factor III. Uso de técnicas y recursos didácticos.

Valora qué tan frecuentemente el profesor utiliza material didáctico o tecnológico para la enseñanza, propone actividades extra clase para reforzar el aprendizaje, imparte la clase de forma dinámica, emplea diversas técnicas pedagógicas para ello y propone temas actuales para discutir en clase.

1.3 Situación actual de la Evaluación Docente en la Institución Educativa

En la actualidad, la evaluación docente forma parte fundamental dentro de los programas y proyectos que orientan los procesos dentro de las instituciones educativas, ya sea por contar con un mecanismo que permita la certificación de la actividad docente, por contar con procedimientos confiables y objetivos de selección y promoción de docentes, para encontrar áreas de mejora y oportunidades en esta labor o por la importancia que cada institución le da a la docencia.

Existen diversas características que conforman la función del docente y en cuanto a evaluación, también hay diferentes perspectivas para realizarlo, debidas entre otras cosas, a la falta de un concepto claro que integre el conjunto de habilidades, conocimientos técnicos, pedagógicos y científicos que un docente debe de poseer.

La evaluación dentro del proceso educativo es relativamente reciente. Su práctica se ha vuelto necesaria dentro de una gama de transformaciones sociales, culturales y pedagógicas, donde se ha creado la necesidad de garantizar el buen funcionamiento de los procesos de enseñanza aprendizaje.

A partir de diferentes valoraciones sobre quién es el docente, en qué institución educativa trabaja, con quién trabaja y cuál es su función, se han generado distintos enfoques desde los cuales se le puede evaluar. Con ello también se han contemplado el diseño y construcción de diferentes procesos de evaluación docente que orienten, de acuerdo a cierta perspectiva, el actuar y la función docente, así como el análisis y la valoración de los procesos mismos de evaluación.

Existe una gran variedad de formas de llevar a cabo la evaluación docente en espacios y momentos concretos. Actualmente la evaluación docente es generalmente promovida como motor de mejora. Pero se tiene una concepción de la evaluación como algo estático, cuya finalidad es identificar “buenos y malos”

docentes a través de instrumentos que permitan una recogida confiable de información sobre el desempeño de los docentes en las aulas.

En la actualidad el carácter de la evaluación docente es principalmente formativo y está orientado a mejorar la labor pedagógica de los educadores y a promover su desarrollo profesional continuo. En vez de ser la pauta para toma de decisiones en cuanto a la permanencia en la docencia de los profesores, las evaluaciones ayudan a:

- Localizar aspectos a mejorar e identificar áreas que requerirán mayor atención, cuando aún se está en posibilidad de remediarlas.
- Sugerir y planificar estrategias que ayuden a superar los aspectos más débiles.
- Recolectar evidencia sobre el avance o progreso.
- Ayudar a evaluar su propio progreso lo que, a su vez, lo alienta a mejorar sus áreas débiles.
- Mejorar los mecanismos que apoyan el proceso de aprendizaje.
- Conocer la Evaluación Docente y asumirla como un proceso positivo que contribuirá a mejorar prácticas pedagógicas y elevar la calidad de la Educación.
- Valorar del sistema de evaluación docente y su estrecha relación con el marco para la buena enseñanza

Actualmente la evaluación docente se lleva a cabo en la mayoría de las instituciones educativas a partir de llenar encuestas a mano por parte de los alumnos en las últimas semanas de cada curso. Estas evaluaciones las lleva a cabo personal administrativo o aplicadores. Los aplicadores son los encargados de llevar a cabo el proceso de evaluación. Antes de la aplicación deben verificar todo el material necesario: gafete de presentación, hojas de respuestas, cuestionarios, carteles de identificación, bitácoras, sobres, lápices e información con horarios y salones. Además deberán revisar las preguntas frecuentes y recordar sus respuestas para resolver dudas de los alumnos. Además durante la

aplicación se debe solicitar el tiempo al profesor y pedirle que se retire durante la actividad, ya que esta se lleva a cabo durante el horario de clases.

En algunas instituciones de educación superior se ha diseñado un cuestionario para cada carrera, mientras que en otras se ha hecho un análisis comparativo de los distintos instrumentos para identificar las coincidencias y a partir de ellas generar una serie de preguntas para conformar un nuevo cuestionario unificado.

En otras instituciones la evaluación se lleva a cabo mediante entrevistas a los alumnos, estos alumnos se eligen de manera aleatoria y el aplicador va tomando nota de un cuestionario previamente preparado.

La ausencia tanto de profesores como alumnos en el momento de la visita de los aplicadores al grupo es un obstáculo para poder realizar la evaluación, ya que no se pueden obtener datos fidedignos si el grupo no está completo o si los alumnos elegidos para entrevistas no representan en su totalidad la perspectiva del grupo hacia el profesor.

El sistema actual entrega datos que pueden ayudar a los docentes a mejorar en ciertos aspectos de su desempeño, aunque también se puede observar que algunos datos estadísticos no son tan claros para extraer conclusiones de las fortalezas de los docentes. También es fácil notar que con estos procesos se manejan gastos que podrían tomarse como innecesarios.

Una vez realizado el proceso, los profesores deben esperar por lo menos un semestre, para que los resultados de la evaluación sean entregados.

Un sistema en línea podría ayudar a evadir varias problemáticas, trabajando de manera eficiente tanto autoridades de la institución educativa como alumnos en el proceso de evaluación, para así poder obtener información concisa y clara.

1.5 Organigrama de la Institución y responsabilidades de cada área

El objetivo de éste inciso es proporcionar información de las funciones de las diversas áreas que consideramos que comúnmente integran alguna Institución Educativa, al personal, autoridades universitarias y a la comunidad en general, que sirva como medio de consulta y les permita conocer las interrelaciones y responsabilidades de las áreas de servicio, facilitando así su interacción.

DIRECCIÓN: Dirigir las actividades académicas y administrativas encaminadas a la formación de profesionales e investigadores de la más alta calidad académica y científica, para dar respuesta a la demanda de la sociedad, con una actitud humanista y comprometidos con principios y valores éticos.

SECRETARIA DE LA DIRECCIÓN: Coordinar y colaborar en las tareas encomendadas por la Dirección para facilitar la gestión de los asuntos académicos y administrativos requeridos por la comunidad interna y externa.

SECRETARÍA GENERAL: Dirigir y coordinar las actividades académicas relacionadas con las funciones de docencia, investigación y extensión de la cultura que la facultad tiene encomendadas para el óptimo desarrollo de las mismas.

SECRETARÍA ACADÉMICA: Establecer las condiciones óptimas de operación para que las actividades académico–administrativas se desarrollen conforme a lo establecido en el proyecto académico.

SECRETARÍA DE ASUNTOS ESCOLARES: Que todos los alumnos desde su ingreso hasta su egreso, cuenten con la información de su situación escolar en todo momento, conozcan sus calificaciones oportunamente, obtengan comprobantes de estudio, posean la información suficiente y bastante sobre los distintos campos en los que pueden desempeñar su servicio social, cuenten con la posibilidad de obtener un empleo, además de que en cualquier momento de su

carrera se les brinde el apoyo que necesiten así como asesorarlos sobre las mejores posibilidades para la conclusión de la misma.

SECRETARÍA DE ASUNTOS ESTUDIANTILES: Difundir, organizar y administrar las diversas actividades académicas, culturales, sociales y de actualización de la carrera.

SECRETARÍA AUXILIAR DE MESA DE FIRMAS: Proporcionar atención y asesoría a los académicos de la Facultad, que lo requieran.

COORDINACIÓN DE LA BIBLIOTECA:

- Proporcionar los servicios bibliotecarios que requiere la comunidad académica, para apoyar la docencia, investigación y difusión de la cultura.
- Desarrollar las colecciones documentales necesarias que atiendan las necesidades de los usuarios.
- Contribuir al desarrollo de habilidades informativas del personal docente y de los alumnos.

SECRETARIA AUXILIAR DE EXÁMENES PROFESIONALES: Realizar los trámites necesarios para que los alumnos que han concluido sus estudios presenten su examen profesional, siempre que hayan optado por cualquiera de estas tres modalidades de titulación: a) Tesis y examen profesional, b) Totalidad de Créditos y Alto Nivel Académico y Examen General de Conocimientos. Lo anterior con el fin de que obtengan su título y cédula profesionales.

SECRETARÍA DE PROGRAMACIÓN DE HORARIOS: Programar los horarios cada semestre.

DIVISIÓN DE UNIVERISDAD ABIERTA Y EDUCACION A DISTANCIA: Elaborar, revisar, actualizar y dar seguimiento a los exámenes tipo y Bancos de Reactivos, Guías de Estudio, Material didáctico, Videoconferencias y a los Paneles de temas de actualidad y que utilizan los Tutores para diagnosticar la calidad de sus métodos de evaluación y adoptar las medidas necesarias para su mejoramiento en beneficio de sus alumnos y lograr la excelencia educativa.

SECRETARÍA DE EDUCACIÓN A DISTANCIA: Conjugar los esfuerzos de profesores y alumnos que están inmersos en una novedosa modalidad de estudio, en el que se utilizan las nuevas tecnologías de información y comunicación, en donde a través de un plataforma montada en internet se interactúa con actividades de aprendizaje, chat, foros y videoconferencias.

DIVISIÓN DE EDUCACIÓN CONTINUA: Ofrecer y promover programas de actualización profesional, capacitación y educación permanente; comprende las actividades educativas relacionadas con la ampliación de conocimientos, la actualización y la capacitación que propicia la superación profesional, así como dar respuesta oportuna y de calidad a las cambiantes exigencias de la práctica profesional, diversificando la oferta educativa.

DEPARTAMENTO DE APOYO ACADÉMICO: Apoyar en la organización de los actos académicos que realiza la División de Educación Continua, formular propuestas para la difusión e instrumentación de los mismos, así como identificar necesidades de formación y actualización de los participantes.

DIVISIÓN DE ESTUDIOS DE POSGRADO: Supervisar, coordinar y ejecutar el programa de estudio de posgrado.

COORDINACIÓN DE ESPECIALIZACIONES: Coordinar y supervisar el Programa de Especializaciones.

SECRETARÍA DE ESTUDIOS DE POSGRADO: Coordinar y supervisar el programa de maestría.

SECRETARÍA TÉCNICA DE PLANEACIÓN: Garantizar la integridad y disponibilidad de información en el apoyo a la toma de decisiones estratégicas, efectivas y rápidas en el momento oportuno. Monitoreo del ambiente académico, evolución y trascendencia de los programas de trabajo conforme a las líneas establecidas en el Plan de Desarrollo Institucional.

COORDINACIÓN DE ENLACE INSTITUCIONAL: Servir como instancia de vinculación entre el cuerpo docente y la División en lo concerniente a la aplicación y calificación de exámenes, y la entrega y seguimiento de los mismos servir de enlace en los procedimientos de revisión de exámenes, así como en la formulación de las actas de rectificación que soliciten los profesores.

DEPARTAMENTO DE CONVENIOS Y REGULACIÓN: Dar seguimiento y actualizar los convenios.

SECRETARIA AUXILIAR DE LA REVISTA DE LA FACULTAD: Enriquecer y difundir el conocimiento jurídico a través de las aportaciones de los maestros, investigadores y especialistas de la Institución y de otras universidades nacionales, así como de instituciones académicas del extranjero.

SECRETARÍA ADMINISTRATIVA: Administrar los recursos humanos, financieros y materiales con que cuenta la dependencia y, proporcionar en forma adecuada y oportuna los servicios de apoyo administrativo requeridos para el cabal cumplimiento de los objetivos, funciones y programas que ésta tiene bajo su cargo.

SECRETARIA DE PERSONAL DOCENTE: Coordinar y supervisar que se proporcionen oportunamente y con calidad los servicios de trámites y prestaciones

para el personal docente con base en las normas y procedimientos administrativos vigentes; además de proporcionar las asesorías y apoyo a las áreas que conforman la Institución.

DEPARTAMENTO DE PERSONAL ADMINISTRATIVO: Coordinar y supervisar que se proporcionen oportunamente y con calidad los servicios de trámites y prestaciones para el personal administrativo con base en las normas y procedimientos administrativos vigentes.

DEPARTAMENTO DE CONTABILIDAD Y PRESUPUESTO: Vigilar, supervisar y controlar los recursos financieros asignados y captados por la facultad, sean administrados y proporcionados oportunamente de acuerdo a la normatividad y procedimientos universitarios; así como atender a las políticas internas para su correcta aplicación.

DEPARTAMENTO DE BIENES Y SUMINISTROS: Proporcionar oportunamente el suministro de bienes, equipos, materiales que satisfagan las necesidades de las áreas de la facultad; así como supervisar y controlar el inventario del activo fijo y de bienes del almacén conforme a la normatividad vigente.

DEPARTAMENTO DE ORGANIZACIÓN Y CONTROL INTERNO: Colaborar y dar apoyo en las actividades y responsabilidades que le confiera el Secretario Administrativo.

DEPARTAMENTO DE SERVICIOS GENERALES: Proporcionar oportunamente los servicios generales que satisfagan las necesidades de las áreas de la Institución.

DEPARTAMENTO DE MANTENIMIENTO Y SERVICIOS DIVERSOS: Llevar a cabo en tiempo y forma los servicios de mantenimiento a vehículos así como

atender las solicitudes de servicios diversos y correspondencia proporcionado a la comunidad.

Todas las áreas anteriores se muestran en la figura 1.4.1.

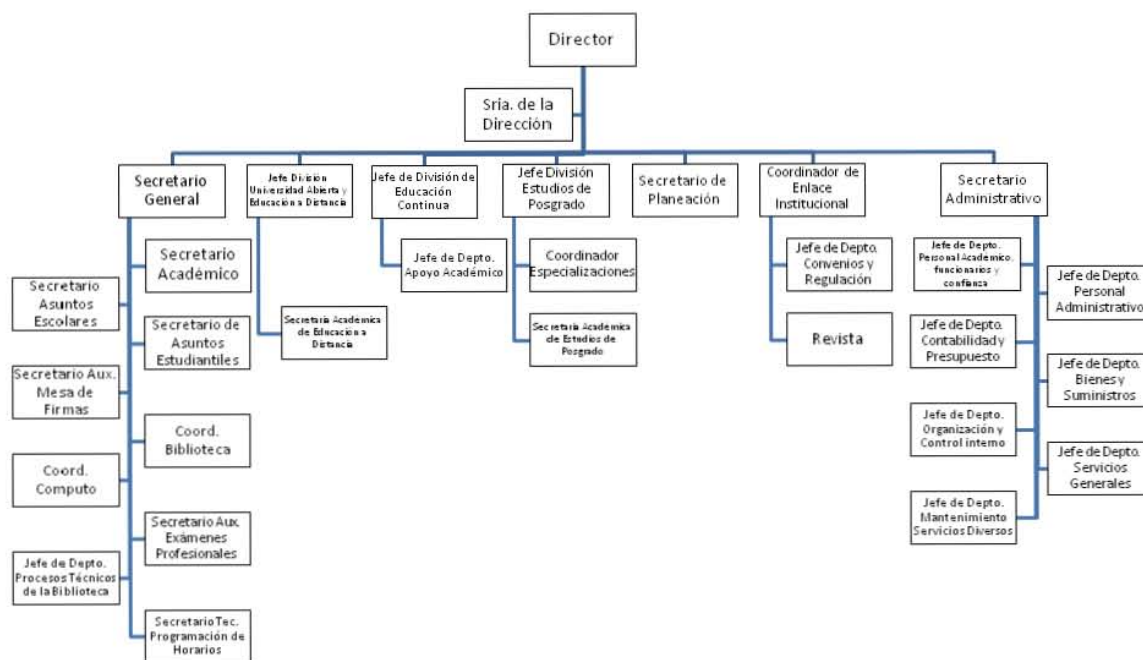


Fig. 1.4.1 Organigrama General

1.6 Metodologías recomendadas pedagógicamente

- **Modelos Pedagógicos**

El proceso pedagógico por sus múltiples funciones y condicionamientos es complejo, necesita ser pensado diseñado con anterioridad de manera que se pueda predecir las modificaciones y transformaciones que propicien su desarrollo. Los modelos son un medio del pensamiento científico, una forma peculiar de abstracción de la realidad, además es casi objeto intermedio auxiliar mediante el cual el conocimiento parece ser trasladado temporalmente del objeto que nos interesa a la investigación.

Es una construcción teórico formal que fundamentada científica e ideológicamente e interpreta, diseña y ajusta la realidad pedagógica que responde a una necesidad histórica concreta.

Un modelo pedagógico es una forma de concebir la práctica del proceso de enseñanza-aprendizaje que consta de varios elementos distintivos. Entre ellos se señala una concepción de cuál es el fin de la educación, un pre supuesto sobre lo que es el alumno, una forma de considerar al profesor, una concepción de lo que es el conocimiento y a su vez una forma de concretar la acción de enseñanza aprendizaje

Hay que comprender que los modelos son construcciones mentales, pues casi la actividad esencial del pensamiento humano a través de su historia ha sido la modelación; y en este sentido construir desde estas visiones estructuradas procedimientos para la enseñanza.

Es importante reglamentar y normalizar el proceso educativo, definiendo ante todo que se debería enseñar, a quienes, con que procedimientos, a qué horas, bajo

que reglamentos disciplinarios, para moldear ciertas cualidades y virtudes en los alumnos.

En este mismo orden de ideas, también plantea algunos ejemplos de modelos, y expresa que existe, el tradicional, conductista, romanticismo, desarrollismo, socialista, y que los define conceptualmente desde la enseñanza de las ciencias.

- **Modelo Pedagógico Tradicional**

El método en el que se hace énfasis es la "formación del carácter" de los estudiantes y moldear por medio de la voluntad, la virtud y el rigor de la disciplina, el ideal del humanismo y la ética, que viene de la tradición metafísica-religiosa del Medioevo.

- **Modelo Transmisionista Conductista.**

El modelo se desarrolló paralelo con la racionalización y planeación económica de los cursos en la fase superior del capitalismo, bajo la mirada del moldeamiento meticuloso de la conducta "productiva" de los individuos. Su método consiste en la fijación y control de los objetivos "instruccionales" formula con precisión. Se trata de una "transmisión parcelada de sus saberes técnicos mediante un adiestramiento experimental "por medio de la "tecnología educativa"

- **Desarrollismo Pedagógico**

Hay una meta educativa, que se interesa porque cada individuo acceda, progresivamente, a la etapa superior de desarrollo intelectual, de acuerdo con las necesidades y condiciones de cada uno. Por otro lado el docente debe crear un ambiente estimulante de experiencias que le permitan al alumno su acceso a la estructura cognoscitiva de la etapa inmediatamente superior.

- **Modelo Pedagogía Socialista (crítico)**

Su pretensión gira en torno al desarrollo máximo y multifacético de las capacidades e intereses del individuo. Este desarrollo es determinado por la sociedad, por la colectividad en la cual el trabajo productivo y la educación son inseparables, y ello garantiza no sólo el desarrollo del espíritu colectivo sino que también el conocimiento pedagógico polifacético y politécnico y el fundamento de la práctica para la formación científica de las nuevas generaciones.

Sus objetivos podrían materializarse interrelacionando y proponiendo formas que permitan introducir este avance tecnológico en las actividades y modalidades educacionales.

La apuesta quedaría focalizada en la elaboración de propuestas innovadoras y creativas que permitan la utilización de estos medios en concordancia con postulados teóricos e ideas conceptuales contemporáneas propias de la epistemología pedagógica.

Emergerían como directrices fundamentales y fundantes – cualesquiera sea la modalidad de enseñanza-, en el desarrollo de modelos pedagógicos para las actividades educativas, elementos como: currículos flexibles, aprendizaje social, significativo y abierto, centrar el proceso en quien aprende, aprender a aprender respeto a ritmos individuales, igualdad, equidad, evaluación permanente y auténtica, comunicabilidad de los medios y mediación didáctica.

En este sentido, el factor principal de todo programa educativo, que considere los aspectos anteriormente mencionados, ha de ser la presencia de un modelo pedagógico que lo sustente, constituyéndose así, en su principio generador al que se subordinarán las estrategias de aprendizaje, la forma de entrega de contenidos, los procesos cognitivos, las interacciones pedagógicas, la concepción evaluativo y

los perfiles que caracterizan a los agentes participantes (profesores, tutores y estudiantes).

El discurso pedagógico precedente y su transferencia a la educación a distancia, tal como la hemos entendido hasta aquí, requiere ser analizado y profundizado teniendo en consideración los siguientes elementos:

Las teorías de aprendizaje, presentes en el modelo deben dar cuenta del cómo es entendido el fenómeno de aprender por la organización que ofrece los cursos.

Por un lado, estaría el problema de compatibilizar el hecho que el estudiante a distancia aprende en "soledad" las teorías contemporáneas del constructivismo social. En consecuencia, facilitar la comunicación y el diálogo, de tal forma que se permita compartir, intercambiar y discutir aprendizajes con los demás agentes educativos.

Por otro lado, posibilitar el "meta aprendizaje" capaz de generar una auténtica reflexión cognitiva. Para tal efecto, deben privilegiarse actividades donde el aprendizaje se constituya en conocimiento aplicado contextualizado a las realidades propias de los docentes.

El modelo pedagógico no sólo impone como exigencia consignar el modo en que se han de entregar dichos contenidos a los estudiantes, sino también, cuál será la participación de ellos en la búsqueda o recepción de la información.

A tal fin, de acuerdo al nivel de los estudiantes, los cursos deberían tener una marcada tendencia hacia el descubrimiento e indagación por iniciativa individual y/o colaborativa.

Aún cuando el diseño e instrucción a la entrega de un modo preferente de organización y construcción de conocimientos, es en la programación didáctica de

cada curso donde quedarían establecidas, mediante la estructuración de un currículo flexible, las condiciones que permitan un aprendizaje abierto.

- **Roles y tareas acordes con el modelo pedagógico.**

Resulta imprescindible definir y apropiarse de nuevos perfiles y características específicas para los profesores, tutores y estudiantes en cuanto son éstos quienes en definitiva construyen el proceso educativo. El profesor planifica y diseña las experiencias y actividades necesarias para la adquisición de los aprendizajes previstos; define los espacios y recursos adecuados para su logro; pero, por sobre todo, asume en su calidad de tutor, un rol facilitador de los aprendizajes, haciendo que el alumno profundice en sus conocimientos, descubra la relevancia que éstos tienen y logre los objetivos propuestos.

A su vez, el alumno juega un papel activo, desarrollando el plan de acción que el profesor le proporciona, excepcionando las indicaciones que lo orienten y ayuden a salvar los obstáculos inherentes a las dificultades propias de todo proceso de enseñanza – aprendizaje.

Como elaborar los modelos pedagógicos

- Las concepciones o teorías filosóficas y sociológicas que les sirven de base general.
- Las teorías psicológicas que les sirven para abordar el papel y funciones de los componentes personales en el proceso de enseñanza - aprendizaje.
- Las teorías pedagógicas que les permiten estructurar las relaciones objetivos - contenidos - métodos– medios y evaluación de la enseñanza y el aprendizaje.

Naturalmente en cualquiera de los modelos pedagógicos pueden encontrarse con mayor o menor claridad los fundamentos filosóficos, psicológicos y pedagógicos

en que se asientan, como también pueden realizarse generalizaciones donde se hace abstracción de las diferencias no esenciales entre unos y otros para agruparlos según sus aspectos más generales.

En este último sentido es posible elaborar una caracterización de dichos modelos, que nos distinguiría dos grandes grupos ubicados en la llamada concepción "Tradicionalista" o en la concepción "Humanista". Dentro de cada uno de los grupos nos quedarán ubicadas las muy diversas variantes de modelos educativos y pedagógicos conocidos.

Para la concepción Tradicionalista, también llamada "externalista" o Escuela Pasiva, la personalidad es el resultado de la influencia de factores externos al sujeto: el maestro, la familia, el medio social y el grupo. El sujeto tiene un papel pasivo, como asimilador y reproductor de esas influencias positivas o negativas, por lo que la enseñanza debe seleccionar aquellas de carácter beneficioso y organizar a todos los agentes socializadores para la acción sobre el sujeto, de lo que cabe esperar un resultado positivo, medible en cuanto al grado en que el sujeto reproduce las influencias recibidas.

Para la concepción Humanista, también llamada "desarrolladora" o Escuela Activa, el sujeto ocupa el primer plano dentro de todo el fenómeno educativo y del proceso pedagógico. Los factores internos de la personalidad se reconocen como elementos activos de la educación del sujeto, en particular sus motivaciones, a la vez que se admite la variedad de respuestas posibles ante las mismas influencias externas.

Desde esta concepción el sujeto se autoeduca mediante la recreación de la realidad, participa en ella y la transforma. Por esta razón la enseñanza – aprendizaje debe ponerse en función de las necesidades individuales y no puede aspirar a la reproducción de un modelo único de individuo sino a la combinación de la socialización y la individualización del sujeto de la manera más plena posible.

Dentro de la Pedagogía Tradicionalista o externalista se pueden ubicar todos los modelos educativos pedagógicos que, partiendo de una base filosófica idealista asumen los métodos de la escolástica medieval, perceptibles en muchas de las prácticas pedagógicas que aún subsisten en las escuelas. También pueden incluirse en este grupo las teorías pedagógicas conductistas, encaminadas a "formar al sujeto" según el deseo del maestro, o las derivadas del pragmatismo, preocupadas esencialmente del resultado final de la enseñanza como reproducción del conocimiento considerado valioso.

Lo que se debate actualmente es el paso de la "Pedagogía del Saber" (expresión de la concepción tradicionalista) hacia la "Pedagogía del Ser", que constituiría el resumen de la mejor tradición humanista en el campo de la educación y la enseñanza.

Dicha Pedagogía del ser tiene como objetivo la felicidad del hombre, su educación para la vida plena, su integración armónica al contexto social desde una perspectiva personal y creadora, en oposición a una Pedagogía del saber aún dominante, que se preocupa por asegurar la repetición de las normas creadas, de los saberes acuñados por otros, de la enajenación del sujeto individual en función de supuestos intereses sociales o grupales que no siempre tienen igual significación para los individuos, puesto que, en última instancia no han sido elaborados por ellos mismos.

Tipos de Modelos

- **Modelo Icónico**

Es una reproducción a escala del objeto real y sus propiedades relevantes. El modelo muestra la misma figura, proporciones y características que el objeto original.

Por ejemplo, se puede construir un modelo a escala de la estructura de un aula, de una institución universitaria. Inclusive estos modelos los podemos someter a determinadas transformaciones para estudiar la funcionalidad del aula o de la universidad.

- **Modelo Analógico**

Es un modelo con apariencia física distinta al original, pero con comportamiento representativo. El modelo analógico no es una reproducción detallada de todas las cualidades del sistema real, sino que refleja solamente la estructura de relaciones y determinadas propiedades fundamentales de la realidad. Se establece una analogía entre el sistema real y el modelo, estudiándose el primero, utilizando como herramienta auxiliar el segundo.

Por ejemplo, en el campo de la Psicología, la conducta de aprendizaje de los animales (ratas, perros, monos, etc.), ha servido como modelo analógico para estudiar las leyes del aprendizaje humano.

Otro ejemplo, lo encontramos en las computadoras electrónicas, las que han servido como modelos materiales de las operaciones intelectuales del hombre.

- **Modelo Teórico**

El modelo teórico (también denominado por algunos autores, analítico) utiliza símbolos para designar las propiedades del sistema real que se desea estudiar. Tiene la capacidad de representar las características y relaciones fundamentales del fenómeno, proporcionar explicaciones y sirve como guía para generar hipótesis teóricas.

Generalmente se considera que revelan relaciones matemáticas o lógicas que representan leyes físicas que se cree gobiernan el comportamiento de la situación bajo investigación.

Es un ejemplo de modelo teórico, la concepción de L. S. Rubinstein, que postula el análisis, la síntesis, la abstracción y la generalización como las operaciones básicas del pensamiento. Frecuentemente los símbolos y fórmulas de la Matemática y la Lógica son utilizados para la elaboración de los modelos teóricos, como lo encontramos en la teoría del pensamiento de Jean Piaget.

De igual forma, la Cibernética como teoría general de la dirección, se ha utilizado como un modelo teórico para explicar el proceso de enseñanza aprendizaje, ya que la enseñanza puede considerarse como una determinada forma de dirección del desarrollo de los procesos psíquicos de la personalidad.

Características de los Modelos

El concepto de modelo pedagógico incluye tres dimensiones, subsistemas o componentes básicos, a saber:

- **Un subsistema teórico.** Este incluye los paradigmas asumidos; los fundamentos filosóficos, epistémicos, psicológicos, pedagógicos, sociológicos e investigativos asumidos como referentes; los conceptos, regularidades y principios pedagógicos/didácticos tomados en consideración y los aportados por el investigador.
- **Un subsistema metodológico.** Se refiere al contenido del modelo que da respuesta a los aspectos teóricos en que se sustenta, declarando las posiciones respecto a componentes personales y no personales, sus etapas y sus relaciones. Por ende, hace alusión al docente, los objetivos pedagógicos, los contenidos de enseñanza - aprendizaje, etc.

- **Un subsistema práctico.** Este prevé el planeamiento y la corroboración práctica del modelo, su instrumentación o implementación, el camino para su concreción en la práctica pedagógica y con ello el logro de la transformación esperada.

La dimensión práctica de un modelo implica la clarificación de los fines a alcanzar, las premisas para ello y las fases o etapas para su obtención. La implementación de un modelo puede asumir diferentes formas: estrategias, metodologías, alternativas didácticas/pedagógicas, programas de intervención educativa y otras.

OTROS MODELOS

- **Modelo Didáctico**

Construcción teórico formal que basada en supuestos científicos e ideológicos pretende interpretar la realidad escolar y dirigirla hacia determinados fines educativos.

Representación simbólica conceptual de la realidad educativa, tendrá por objetivo funcionar como esquema mediador entre la realidad educativa y el pensamiento. Sirve como estructura en torno a la cual se organiza el conocimiento

- **Modelo Educativo**

Implica la política educativa, la filosofía de la educación y la concepción teórica sobre educación. Pretende unidad de los códigos culturales y se concreta en la comunidad (participantes del hecho educativo).

- **Modelo Pedagógico**

Implica el contenido de la enseñanza, el desarrollo del alumno y la característica de la práctica del docente. Pretende lograr aprendizajes y se concreta en el aula.

Instrumento de la investigación de carácter teórico creada para reproducir idealmente el proceso enseñanza - aprendizaje. Paradigma que sirve para entender, orientar y dirigir la educación.

Se hace evidente la diversidad de conceptos determinantes asociados a la definición de modelo pedagógico.

Asumir una definición operativa implica declarar de qué pre supuestos teóricos se parten, en los que se deje claro el término pedagógico quien revela la esencia del modelo.

CAPITULO II. MARCO TEÓRICO

CAPITULO II. MARCO TEÓRICO

2.1 Características, ventajas y desventajas de la metodología RUP y lenguaje UML.

En la actualidad, la utilización de metodologías para el desarrollo de aplicaciones es casi imposible omitirlas, debido a la gran necesidad de control de variables que conlleva el mismo desarrollo, y para la elaboración ordenada de las aplicaciones, por lo tanto, seguir metodologías y estándares nos llevan a estar en competitividad en todo momento.

Es de suma importancia conocer el modo como se interrelacionan metodologías con estándares y herramientas siguiendo un único propósito, el cual consiste en la elaboración de aplicaciones de manera eficiente, ordenada y con el menor número de defectos.

La metodología RUP nos proporciona disciplinas en las cuales se encuentran artefactos con lo cual se podrá contar con guías para poder documentar e implementar de una manera fácil y eficiente, todas las guías para un buen desarrollo, todo esto dentro de las respectivas fases con las cuales cuenta.

Las metodologías y estándares utilizados en un desarrollo de software nos proporcionan las guías para poder conocer todo el camino a recorrer desde antes de empezar la implementación, con lo cual se asegura la calidad del producto final, así como también el cumplimiento en la entrega del mismo en un tiempo estipulado.

Es de suma importancia elegir la metodología, así como las herramientas de implementación adecuadas, es por ello que la metodología RUP basada en UML nos proporciona todas las bases para llevar al éxito la elaboración del software.

Rational Unified Process (RUP)



Figura 2.1.1 Logo de Metodología RUP

Las siglas RUP en inglés vea logo la figura 2.1.1 significa Rational Unified Process (Proceso Unificado de Racional) es un producto del proceso de ingeniería de software que proporciona un enfoque disciplinado para asignar tareas y responsabilidades dentro de una organización del desarrollo. Su meta es asegurar la producción del software de alta calidad que resuelve las necesidades de los usuarios dentro de un presupuesto y tiempo establecidos.

Es una metodología para desarrollo de software, para el análisis y diseño, es un proceso basado en componentes, dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental. requiere mucha disciplina para su implementación ya que inicialmente se debe ajustar la metodología a las necesidades del proyecto, equilibrando los esfuerzos, talento humano y tiempos para llevar a cabo en los mejores términos dicho proyecto.

Esta metodología busca constantemente calidad en sus procesos, permitiendo fortalecer el proyecto y superando las expectativas del cliente con los ejercicios iterativos en las etapas del proyecto, así se va refinando el producto que va ser entregado al usuario final buscando que realmente el producto satisfaga plenamente lo que necesita el usuario; estas iteraciones son desarrolladas por el trabajo en equipo el cual está conformado por diferentes personas con diferentes funciones (analistas, diseñadores, administrador base de datos, testers, arquitecto

de datos, etc.) quienes deben mantener una buena comunicación para mantener a todo el equipo al tanto del desarrollo del proyecto.

Un aspecto importante que permite llevar a cabo con satisfacción la implementación de la metodología RUP (como en cualquier metodología de software) es la documentación técnica y metodológica del proyecto, en primera instancia la aplicación de ingeniería de requerimientos, diagramas bajo el lenguaje de modelado UML (casos de uso, diagramas de casos de uso), documentos de los procesos metodológico del proyecto, pruebas, entregables, etc.

El proceso unificado actúa como un modelo que puede adaptarse a cualquier tipo de proyecto y empresa (grandes y pequeñas).

Las características del proceso unificado de modelado son:

- **Centrado en los Modelos:** Los diagramas son un vehículo de comunicación más expresivo que las descripciones en lenguaje natural. Se trata de minimizar el uso de descripciones y especificaciones textuales del sistema.
- **Guiado por lo casos de uso:** Los casos de uso son el instrumento para validar la arquitectura del software y extraer los casos de prueba.
- **Centrado en la arquitectura:** Los modelos son proyecciones del análisis y el diseño constituye la arquitectura del producto a desarrollar.
- **Iterativo e incremental:** Durante todo el proceso de desarrollo se producen versiones incrementales (que se acercan al producto terminado) del producto en desarrollo.

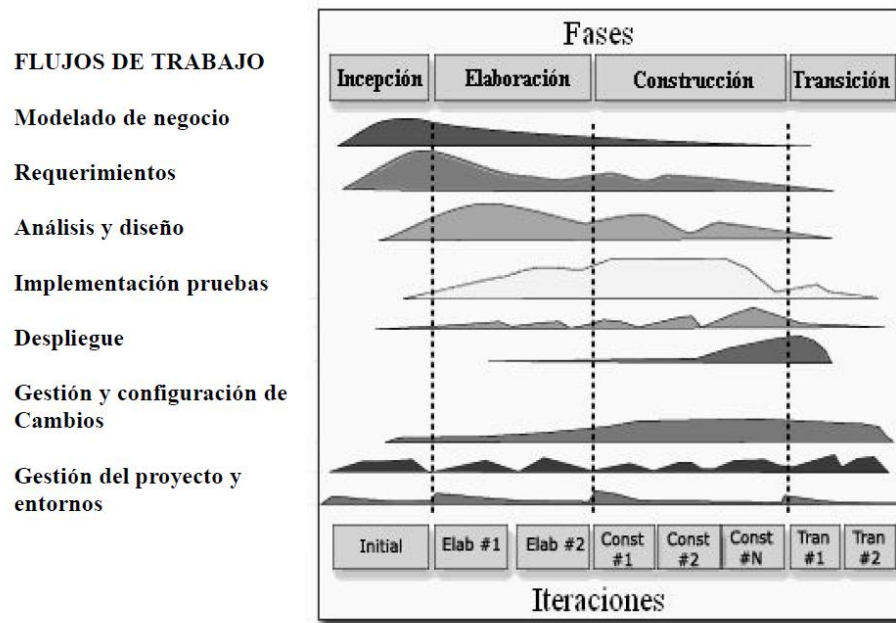


Figura 2.1.2 El Proceso de Modelado Unificado

La figura 2.1.2 que representa el RUP incluye las cuatro etapas importantes que son: la iniciación, elaboración, construcción y transición, las cuales muestran que para producir una versión del producto en desarrollo se aplican todas las actividades de ingeniería pero con diferente énfasis; en las versiones preliminares, como además indica la intuición, hay más énfasis en actividades de modelado del negocio, requisitos, análisis y diseño; conforme se producen versiones el énfasis pasa a las actividades de implementación, pruebas y despliegue.

Características principales de RUP

- **Centrado en los modelos:** Los diagramas son un vehículo de comunicación más expresivo que las descripciones en lenguaje natural. Se trata de minimizar el uso de descripciones y especificaciones textuales del sistema.

- Guiado por los Casos de Uso: Los Casos de Uso son el instrumento para validar la arquitectura del software y extraer los casos de prueba.
- Centrado en la arquitectura: Los modelos son proyecciones del análisis y el diseño constituye la arquitectura del producto a desarrollar.
- Iterativo e incremental: Durante todo el proceso de desarrollo se producen versiones incrementales (que se acercan al producto terminado) del producto en desarrollo.
- Captura varias de las mejores prácticas en el desarrollo moderno de software en una forma que es aplicable para un amplio rango de proyectos y organizaciones.
- Es una guía de cómo utilizar de manera efectiva UML.
- Provee a cada miembro de un equipo un fácil acceso a una base de conocimiento con guías, plantillas y herramientas para todas las actividades críticas de desarrollo.
- Crea y mantiene modelos, en lugar de enfocarse en la producción de una gran cantidad de papeles de documentación.
- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas de Ingeniería de software.
- Desarrollo iterativo.

- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual de software.
- Verificación de de calidad de software.

Ventajas de utilizar RUP

- Permite desarrollar aplicaciones sacando el máximo provecho de las nuevas tecnologías, mejorando la calidad, el rendimiento, la reutilización, la seguridad y el mantenimiento del software mediante una gestión sistemática de los riesgos.
- Permite la producción de software que cumpla con las necesidades de los usuarios, a través de la especificación de los requisitos, con una agenda y costo predecible.
- Enriquece la productividad en equipo y proporciona prácticas óptimas de software a todos sus miembros.
- Permite llevar a cabo el proceso de desarrollo práctico, brindando amplias guías, plantillas y ejemplos para todas las actividades críticas.

- Proporciona guías explícitas para áreas tales como modelado de negocios, arquitectura Web, pruebas y calidad. También se proporciona guías para desarrollar en plataformas IBM WebSphere y Microsoft Web Solution para acelerar el desarrollo de los proyectos.
- Se integra estrechamente con herramientas Rational, permitiendo a los equipos de desarrollo aprovechar todas las ventajas de las características de los productos Rational, el Lenguaje de Modelado Unificado (UML) y otras prácticas óptimas de la industria.
- Unifica todo el equipo de desarrollo de software y mejora la comunicación al brindar a cada miembro del mismo una base de conocimientos, un lenguaje de modelado y un punto de vista de cómo desarrollar software.
- Optimiza la productividad de cada miembro del equipo al poner al alcance la experiencia derivada de miles de proyectos y muchos líderes de la industria.
- No solo garantiza que los proyectos abordados serán ejecutados íntegramente sino que además evita desviaciones importantes respecto a los plazos.
- Permite una definición acertada del sistema en un inicio para hacer innecesarias las reconstrucciones parciales posteriores.
- Esta es una metodología completa en sí misma, con un énfasis en la documentación precisa.

- Es capaz de resolver de forma proactiva los riesgos del proyecto asociado a las nuevas necesidades de los clientes que requieren una cuidadosa gestión de solicitud de cambio.
- Se requiere menos tiempo en el proceso de integración a lo largo del ciclo de vida de desarrollo.
- El tiempo de desarrollo menor debido a la reutilización de componentes.
- Aumento de la productividad.
- Disminución del tiempo y costo de formación y entrenamiento.
- Disminuye el número de errores de usuario.
- Aumento en la exactitud en la entrada y el entendimiento de los datos.
- Disminución de la necesidad de soporte técnico continuo.
- Mayor ingreso por producto o servicio competitivo.
- Disminución de los costos de desarrollo y mantenimiento.
- Disminución de los costos de atención al cliente.
- Más oportunidades de negocio debido a la satisfacción.
- Es utilizada por gran cantidad de desarrolladores que han obtenido resultados exitosos.

- Es adaptable a una amplia gama de tipos de proyectos y clientes.
- Se encuentra constantemente haciendo verificaciones de calidad se apoya en UML lo que permite tener modelado visual.
- Es una metodología ordenada y formal.

Desventajas de utilizar RUP

- Los miembros del equipo deben ser expertos en su campo para desarrollar un software bajo esta metodología.
- El proceso de desarrollo es demasiado complejo y desorganizado.
- En la reducción de los proyectos de vanguardia que utilizan las nuevas tecnologías, la reutilización de componentes no será posible.
- La integración en el proceso de desarrollo de software, en teoría parece una buena cosa. Pero en particular los grandes proyectos de desarrollo con flujos múltiples sólo servirá para aumentar la confusión y causar más problemas durante las etapas de prueba.
- Es una metodología muy rígida.
- Si el conjunto de documentos y artefactos no son concebidos tal y como se plantean en RUP, dicha documentación solo servirá para ser archivada, lo cual no genera valor respecto a la calidad del desarrollo. Y evoluciona en problemas más complejos tales como: aplicaciones que no satisfacen los

requerimientos, o que los diseños de las estructuras no coincidan con las estructuras finales de las aplicaciones, etc.

- Necesita gente con experiencia, especialmente para el diseño.
- Se trata de un proceso de implementación complejo, para el cual el número de gente especializada en el grupo de trabajo debe aumentar con respecto a otras metodologías.
- En RUP se pasa más tiempo que en otras metodologías antes de llegar a la fase de desarrollo.

UML (Unified modeling language) Vea logo Figura 2.1.3



Figura 2.1.3 Logo de UML

UML significa "Unified Modeling Language": Lenguaje de Modelado o Modelamiento Unificado.

El Lenguaje de Modelado Unificado es un lenguaje usado para especificar, visualizar y documentar los diferentes aspectos relativos a un sistema de software bajo desarrollo, así como para modelado de negocios y otros sistemas.

Puede ser utilizado con cualquier metodología, a lo largo del proceso de desarrollo de software, en cualquier plataforma tecnológica de implementación (Unix, Windows etc.).

Es un sistema notacional (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelado que utilizan conceptos orientados a objetos.

Los principales factores que motivaron la definición de UML fueron: la necesidad de modelar sistemas, en la industria del software, unificar los distintos lenguajes y métodos existentes e innovar los modelos para adaptarse a la arquitectura distribuida.

Es importante resaltar que un modelo UML describe lo que supuestamente hará un sistema, pero no dice como implementar dicho sistema.

El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

UML se puede usar para modelar distintos tipos de sistemas: de software, de hardware, y organizaciones del mundo real.

UML es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos. El modelado no es más que la construcción de un modelo a partir de una especificación. Un modelo es una abstracción de algo, que se elabora para comprender ese algo antes de

construirlo. El modelo omite detalles que no resultan esenciales para la comprensión del original y por lo tanto facilita dicha comprensión.

UML es una consolidación de muchas de las notaciones y conceptos más usados orientados a objetos.

Características principales de UML

- UML es una especificación de notación orientada a objetos. Divide cada proyecto en un número de diagramas que representan las diferentes vistas del proyecto. Estos diagramas juntos son los que representa la arquitectura del proyecto.
- UML permite describir un sistema en diferentes niveles de abstracción, simplificando la complejidad sin perder información, para que tanto usuarios, líderes y desarrolladores puedan comprender claramente las características de la aplicación.
- UML se quiere convertir en un modelo estándar con la que sea posible modelar todos los componentes del proceso de desarrollo de aplicaciones. Sin embargo, hay que tener en cuenta un aspecto importante del modelo: no pretende definir un modelo estándar de desarrollo, sino únicamente un lenguaje de modelado.
- En UML los procesos de desarrollo son diferentes según los distintos dominios de trabajo; no puede ser el mismo proceso para crear una aplicación en tiempo real, que el proceso de desarrollo de una aplicación orientada a gestión.
- El método del UML recomienda utilizar los procesos que otras metodologías tienen definidos.

- Combina notaciones provenientes desde: Modelado Orientado a Objetos, Modelado de Datos, Modelado de Componentes, Modelado de Flujos de Trabajo.

Ventajas de utilizar UML

- Es estándar, Facilita la comunicación.
- Está basado en un metamodelo con una semántica bien definida.
- Se basa en una notación gráfica concisa y fácil de aprender y utilizar.
- Se puede utilizar para modelar sistemas de software en diversos dominios: Sistemas de información empresariales, Sistemas WEB, sistemas críticos y de tiempo real, etc. Incluso en sistemas que no son de software.
- Es fácilmente extensible.
- UML es independiente del Proceso de desarrollo.
- Un uso óptimo se consigue en procesos dirigidos por casos de uso, centrados en la arquitectura, iterativos e incrementales Proceso Unificado de Desarrollo (RUP).
- UML cubre las diferentes vistas de la arquitectura de un sistema mientras se evoluciona a través del ciclo de vida del desarrollo de software.
- Proporciona Vistas de Software (estáticas, dinámicas, etc.)

- UML es un lenguaje de propósito general para el modelado orientado a objetos.
- UML ofrece vocabulario y reglas: para crear y leer modelos bien formados, que constituyen los planos de un sistema de software.

Desventajas de utilizar UML

- No es una metodología.
- No cubre todas las necesidades de especificación de un proyecto de software.
- No define los documentos textuales o el diseño de interfaces de usuario.
- Faltan ejemplos elaborados en la documentación.
- Puede resultar complejo alcanzar un conocimiento completo del lenguaje.
- Falta integración con respecto de otras técnicas tales como patrones de diseño, interfaces de usuario, documentación, etc., los ejemplos aislados, el monopolio de conceptos, técnicas y métodos en torno a UML.
- UML no prescribe un proceso o método estándar para desarrollar un sistema.
- El UML es un lenguaje para construir modelos; no una guía para el desarrollador en la forma de realizar el análisis y diseño orientados a objetos ni le indica cuál proceso de desarrollo adoptar.

- UML permite especificar, visualizar y construir softwares, pero orientado a objetos. Para aquellos que prefieran las metodologías estructuradas deberán esperar que surja un Lenguaje Unificado de Modelado Estructurado.

El UML es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos.

Con la creación del UML se persigue obtener un lenguaje que sea capaz de abstraer cualquier tipo de sistema, sea informático o no, mediante los diagramas, es decir, mediante representaciones gráficas que contienen toda la información relevante del sistema.

UML está pensado para el modelado tanto de pequeños sistemas como de sistemas complejos.

En la práctica todos los diagramas son bidimensionales, pero el UML permite crear diagramas en tres dimensiones como en modelos donde se puede "entrar" al modelo para poderlo visualizar por dentro.

DIAGRAMAS

UML recomienda la utilización de nueve diagramas para representar las distintas vistas de un sistema.

Los diagramas de UML son los siguientes:

- **Diagrama de Casos de Uso:** modela la funcionalidad del sistema agrupándola en descripciones de acciones ejecutadas por un sistema para obtener un resultado. Se utiliza para entender el uso del sistema.

Muestra el conjunto de casos de uso y actores (Un actor puede ser tanto un sistema como una persona) y sus relaciones: es decir, muestra quien puede hacer qué y las relaciones que existen entre acciones (casos de uso). Son muy importantes para modelar y organizar el comportamiento del sistema.

- **Diagrama de Clases:** muestra las clases (descripciones de objetos que comparten características comunes) que componen el sistema y cómo se relacionan entre sí.
- **Diagrama de Objetos:** muestra una serie de objetos (instancias de las clases) y sus relaciones. A diferencia de los diagramas anteriores, estos diagramas se enfocan en la perspectiva de casos reales o prototipos. Es un diagrama de instancias de las clases mostradas en el diagrama de clases.
- **Diagrama de Secuencia:** enfatiza la interacción entre los objetos y los mensajes que intercambian entre sí junto con el orden temporal de los mismos.
- **Diagrama de Colaboración:** igualmente, muestra la interacción entre los objetos resaltando la organización estructural de los objetos en lugar del orden de los mensajes intercambiados.

El diagrama de secuencia y el diagrama de colaboración: muestran a los diferentes objetos y las relaciones que pueden tener entre ellos, los mensajes que se envían entre ellos. Son dos diagramas diferentes, que se

puede pasar de uno a otro sin pérdida de información, pero que nos dan puntos de vista diferentes del sistema. En resumen, cualquiera de los dos es un Diagrama de Interacción.

- **Diagrama de Estados:** Se utiliza para analizar los cambios de estado de los objetos. Muestra los estados, eventos, transiciones y actividades de los diferentes objetos. Son útiles en sistemas que reaccionen a eventos.
- **Diagrama de Actividades:** Es un caso especial del diagrama de estados, simplifica el diagrama de estados modelando el comportamiento mediante flujos de actividades. Muestra el flujo entre los objetos. Se utilizan para modelar el funcionamiento del sistema y el flujo de control entre objetos.
- **Diagrama de Componentes:** muestra la organización y las dependencias entre un conjunto de componentes. Se usan para agrupar clases en componentes o módulos.
- **Diagrama de Despliegue (o implementación):** muestra los dispositivos que se encuentran en un sistema y su distribución en el mismo. Se utiliza para identificar Sistemas de Cooperación: Durante el proceso de desarrollo el equipo averiguará de qué sistemas dependerá el nuevo sistema y que otros sistemas dependerán de él.

Los diagramas que utilizaremos para el desarrollo de la presente tesis son:

- **Diagrama de Clases**

En el diagrama de clases será donde definiremos las características de cada una de las clases y relaciones de dependencia y generalización. Este diagrama sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales

pueden ser asociativas, de herencia, de uso y de contenido. En UML, una clase es representada por un rectángulo que posee tres divisiones en la figura 2.1.4

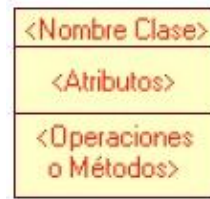


Figura 2.1.4 Representación de una Clase en UML

- **Diagrama de Casos de Usos**

Se emplea para visualizar el comportamiento del sistema, una parte de él o de una sola clase; y como se relaciona con su entorno. De ésta forma se puede conocer cómo responde ésta parte del sistema ante un estímulo del ambiente. El diagrama de casos de uso es muy útil para definir como debería ser el comportamiento de una parte del sistema, ya que solo especifica como deben comportarse y no como están implementadas las partes que define. Un caso de uso especifica un requerimiento funcional.

En la figura 2.1.5 se muestra el diagrama de casos de uso

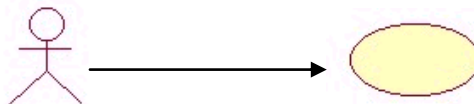


Figura 2.1.5 Diagrama de Casos de Uso

- **Diagrama de Modelado de datos**

Un modelo de datos es un lenguaje orientado a describir una Base de Datos. Típicamente un modelo de datos permite describir:

- Las estructuras de datos de la base: El tipo de los datos que hay en la base y la forma en que se relacionan.
- Las restricciones de integridad: Un conjunto de condiciones que deben cumplir los datos para reflejar correctamente la realidad deseada.
- Operaciones de manipulación de los datos: típicamente, operaciones de agregado, borrado, modificación y recuperación de los datos de la base.

- **Diagrama Modelo entidad-relación**

Un diagrama o modelo entidad-relación (a veces denominado por sus siglas en inglés, E-R "Entity relationship", o del español DER "Diagrama de Entidad Relación") es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

El Modelo Entidad-Relación.

- Se elabora el diagrama (o diagramas) entidad-relación.
- Se completa el modelo con listas de atributos y una descripción de otras restricciones que no se pueden reflejar en el diagrama.

2.2 Características, ventajas y desventajas del lenguaje de programación HTML y PHP.

HTML

En el año de 1962 el concepto de Internet fue concebido por el Departamento de la Defensa de los Estados Unidos. En los primeros años de los noventas el físico inglés Tim Berner-Lee, que trabajaba para el Laboratorio Europeo de Física de Partículas (CERN), propuso una forma de comunicación entre computadoras que facilitaba a los investigadores de todo el mundo conocer los resultados recientes de las investigaciones que ahí se realizaban. Su propuesta es lo que hoy conocemos como HTTP y HTML.

El HTML (Hyper Text Markup Language - Lenguaje de Marcado de Hipertexto) es el lenguaje de programación con el que se escriben las páginas web. Es un lenguaje muy sencillo que permite confeccionar archivos de texto mostrándolo de forma estructurada y atractiva, con enlaces (hyperlinks) que hacen referencia a otros documentos o fuentes de información relacionada, y que permite incorporar archivos multimedia como gráficos, sonidos, imágenes, videos, etc.

Los documentos HTML están formados por un conjunto de etiquetas especialmente intercaladas escritas en texto plano (ASCII). Son interpretadas por los navegadores Web que formatean y muestran los documentos. Un conjunto de documentos HTML pueden estar entrelazados mediante enlaces, en inglés hyperlinks o links.

Los documentos HTML son almacenados en la computadora con la extensión .html o .htm.

HTML consta de varios componentes vitales, entre ellos los elementos y sus atributos, tipos de data y la declaración de tipo de documento.

Un documento HTML, o página web, está formado por etiquetas y cada etiqueta tiene o puede tener atributos. Las etiquetas se identifican fácilmente en una página web, pues se encuentran contenidas entre los símbolos <>.

Características

- Lenguaje estático.
- Basado en etiquetas.
- El estándar lo define el W3C (World Wide Web Consortium).
- Interpretado.
- Extensiones de archivos .html ó .htm.

Ventajas

- Texto estructurado.
- Archivos pequeños.
- Es fácil de entender y utilizar.
- Compatible en todos los exploradores.
- Su uso es muy extendido.

Desventajas

- Estático.
- Interpretación diferente dependiendo del navegador.
- Exceso de etiquetas.
- Tiene un costoso mantenimiento de las páginas.
- No es muy completo.

En resumen, html es un lenguaje muy fácil de comprender y muy utilizado para la presentación de la información, pero esta no se puede procesar ni almacenar, ya que no permite por si solo su manipulación por un programa.

PHP



Figura 2.2.1 Logo PHP

PHP, (acrónimo de "PHP: Hypertext Preprocessor") es un lenguaje "Open Source" interpretado de alto nivel. Especialmente pensado para desarrollos Web y el cual puede ser inmerso en páginas HTML. La mayoría de su sintaxis es similar a C, Java, Perl y es fácil de aprender. El objetivo de este lenguaje es, permitir a los creadores de páginas Web, escribir páginas dinámicas de una manera rápida y fácil. PHP puede procesar la información de formularios, generar páginas con contenidos dinámicos, o enviar y recibir cookies.

PHP es hoy en día el lenguaje interpretado del lado del servidor más utilizado en el desarrollo de aplicaciones web, cuenta con un gran número de colaboradores a nivel mundial, está portado a las plataformas más utilizadas (Windows, GNU/Linux, Mac OSX, etc.), es un proyecto de código abierto y ofrece mucha documentación en su sitio web.

Los scripts desarrollados en PHP se embeben dentro de una página HTML, es común que a los archivos se les coloque la extensión .php en lugar de .html.

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos, lo cual permite la creación de aplicaciones web muy robustas.

Características

Al ser un lenguaje libre dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas web dinámicas:

- Soporte para una gran cantidad de bases de datos: *MySQL, PostgreSQL, Oracle, MS, SQL Server, SybasemSQL, Informix*, entre otras.
- Integración con varias bibliotecas externas, permite generar documentos en PDF hasta analizar código XML.

- PHP es relativamente multiplataforma. Funciona en toda máquina que sea capaz de compilar su código, entre ellas diversos sistemas operativos para PC y diversos Unix. El código escrito en PHP en cualquier plataforma funciona exactamente igual en cualquier otra.
- El acceso a las bases de datos de PHP es muy heterogéneo, pues dispone de un juego de funciones distinto por cada gestor.
- PHP es suficientemente versátil y potente como para hacer tanto aplicaciones grandes que necesiten acceder a recursos a bajo nivel del sistema como pequeños scripts que envíen por correo electrónico un formulario relleno por el usuario.

Ventajas:

- La principal ventaja se basa en ser un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos.
- Posee una muy buena documentación en su página oficial.
- Es Libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos. (Clases y herencia.).
- Muy sencillo de aprender.
- Similar en sintaxis a C y a PERL.
- El análisis léxico para recoger las variables que se pasan en la dirección lo hace PHP de forma automática. Librándose el usuario de tener que separar las variables y sus valores.
- Se puede incrustar código PHP con etiquetas HTML.

- Excelente soporte de acceso a base de datos.
- La comprobación de que los parámetros son validos se hace en el servidor y no en el cliente (como se hace con *javascript*) de forma que se puede evitar que chequear que no se reciban solicitudes adulteradas. Además PHP viene equipado con un conjunto de funciones de seguridad que previenen la inserción de órdenes dentro de una solicitud de datos.
- Se puede hacer de todo lo que se pueda transmitir por vía HTTP.

Desventajas:

- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y php.
- Promueve creación de código desordenado y con un mantenimiento complejo.
- No posee adecuado manejo de unicode.
- Es muy difícil de optimizar.

HTML un día comenzó como un lenguaje para compartir documentos científicos a través de la Internet y es actualmente un estándar abierto. Por otra parte, PHP es el lenguaje más utilizado para desarrollar aplicaciones web de manera sencilla y rápida.

PHP es un lenguaje script procesado en el lado del servidor. El PHP se muestra como código embebido dentro de una página HTML. Este tipo de iteración permite algunas operaciones complejas como conexiones a bases de datos o ejecución de complejos programas.

2.3 Características, ventajas y desventajas del lenguaje de programación JavaScript.

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas.

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

La característica principal de Javascript es la de ser un *lenguaje de scripting*, pero, sobre todo, la de ser el lenguaje de scripting por excelencia y, sin lugar a dudas, el más usado. Esta particularidad conlleva una notable serie de ventajas y desventajas según el uso que se le deba dar y teniendo en cuenta la relación que se establece entre el mecanismo cliente-servidor.

El servidor envía los datos al cliente y estos datos pueden llegar en en dos formatos diversos: en formato texto (o ASCII) o en formato binario por lo que si los datos llegan en este formato son inmediatamente ejecutables, mientras que si el formato es diverso tienen que ser interpretados y traducidos al formato binario y, por tanto, el cliente necesitará un filtro, o mejor dicho, un intérprete que sepa leer estos datos y los pueda traducir al binario. Los datos en formato texto son visibles al usuario como simples combinaciones de caracteres y de palabras y son, por tanto, fáciles de manipular, si bien requieran más tiempo para su interpretación a causa de los distintos pasos y de las transformaciones a las que deben someterse para que el cliente pueda comprenderlos: Los datos en formato binario, sin

embargo, son difíciles de comprender por el usuario, pero inmediatamente ejecutables por el cliente ya que no requieren fases intermedias.

Características:

- Es simple, no hace falta tener conocimientos de programación para poder hacer un programa en JavaScript.
- Maneja objetos dentro de nuestra página Web y sobre ese objeto podemos definir diferentes eventos. Dichos objetos facilitan la programación de paginas interactivas, a la vez que se evita la posibilidad de ejecutar comandos que puedan ser peligrosos para la maquina del usuario, tales como formateo de unidades, modificar archivos etc.
- Es dinámico, responde a eventos en tiempo real. Eventos como presionar un botón, pasar el puntero del mouse sobre un determinado texto o el simple hecho de cargar la página o caducar un tiempo. Con esto podemos cambiar totalmente el aspecto de nuestra página al gusto del usuario, evitándonos tener en el servidor un página para cada gusto, hacer cálculos en base a variables cuyo valor es determinado por el usuario, etc.

La integración de JavaScript y XHTML es muy flexible, ya que existen al menos tres formas para incluir código JavaScript en las páginas web:

- Incluir JavaScript en el mismo documento XHTML.

El código JavaScript se encierra entre etiquetas `<script>` y se incluye en cualquier parte del documento

- Definir JavaScript en un archivo externo

Las instrucciones JavaScript se pueden incluir en un archivo externo de tipo JavaScript que los documentos XHTML enlazan mediante la etiqueta `<script>`. Se pueden crear todos los archivos JavaScript que sean

necesarios y cada documento XHTML puede enlazar tantos archivos JavaScript como necesite.

- Incluir JavaScript en los elementos XHTML

Este último método es el menos utilizado, ya que consiste en incluir trozos de JavaScript dentro del código XHTML de la página.

Ventajas

- Es un lenguaje sencillo.
- Es un lenguaje de programación muy liviano.
- Se recomienda para la creación de aplicaciones web.
- Utiliza poca memoria
- Útil para el desarrollo de páginas web dinámicas.
- Tiene gran cantidad de efectos visuales.
- Fácil manejo de datos.
- Es soportado por los más populares navegadores.
- Puede ser usado en los servidores para hacer guiones de alta calidad.
- Ligero de carga.
- Fácil de integrar.
- Cientos de aplicaciones disponibles para su uso.
- Puede agregar interactividad a elementos web.

Desventajas

- Sus recursos no son tan extensos.
- Sus opciones 3d son limitadas.
- En sistemas no tan complejos puede generar errores.

- Los usuarios pueden desactivar JavaScript en su navegador.
- Un problema importante es que el código es visible y puede ser leído por cualquiera, incluso si está protegido con las leyes de copyright.

A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java.

Desde su aparición, JavaScript siempre fue utilizado de forma masiva por la mayoría de sitios de Internet. La aparición de Flash disminuyó su popularidad, ya que Flash permitía realizar algunas acciones imposibles de llevar a cabo mediante JavaScript.

Sin embargo, la aparición de las aplicaciones AJAX programadas con JavaScript le ha devuelto una popularidad sin igual dentro de los lenguajes de programación web.

2.4 Características, ventajas y desventajas del modelo de Bases de datos relacionales.

Una base de datos relacional es una base de datos en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores (también se les llama relaciones), y en donde todas las operaciones de la base de datos operan sobre estas tablas.

Las tablas se presentan gráficamente como una estructura rectangular formada por filas y columnas. Cada columna almacena información sobre una propiedad determinada de la tabla (se le llama también atributo), como nombre, id, apellidos, edad, etc.

Cada fila de una tabla representa un conjunto de datos relacionados (a las filas se les llama también tuplas), y todas las filas de la misma tabla tienen la misma estructura.

Toda tabla debe tener un campo que sirva para identificar cada uno de sus registros, ese campo es la clave principal. La clave principal proporciona un valor único para cada fila de la tabla y nos sirve de identificador de registros de forma que con esta clave podamos saber sin ningún tipo de equivocación el registro al cual identifica. No podemos definir más de una clave principal, pero podemos tener una clave principal compuesta por más de un campo.

Un campo definido como clave principal no podrá aceptar valores duplicados (es decir que no podrán existir dos filas con el mismo valor en ese campo), ni podrá contener el valor nulo.

Una clave foránea es una columna (o combinación de columnas) que contiene un valor que hace referencia a una fila de otra tabla (en algunos casos puede ser la misma tabla).

Una tabla tiene una única clave primaria. Una tabla puede contener cero o más claves foráneas.

Características

Las bases de datos relacionales tienen algunas características que las distinguen de otros tipos de bases de datos. Son las siguientes:

- Se basa en el álgebra relacional.
- Una base de datos relacional consta de una o más tablas de dos dimensiones de valores de datos, y la construcción de dichas tablas viene definida por reglas muy simples. Cada tabla corresponde a un tipo de entidad conceptual, que no contenga grupos de repetición.
- Las relaciones entre dos o más tablas se establecen en virtud de los valores de datos comunes contenidos en las tablas correspondientes.
- Se ha desarrollado una metodología sistemática para la transformación de un conjunto de tipos de entidad (que representan un diseño conceptual) en un conjunto correspondiente de tablas. El objetivo de esta metodología es la generación de un grupo de tablas en el cual se minimice la duplicación de datos, y en el que se hayan eliminado determinados problemas asociados con el mantenimiento de las tablas.

- Los sistemas de bases de datos relacionales suelen tener asociados lenguajes de consulta de alto nivel, que facilitan la realización de búsquedas y de actualizaciones en la base de datos, de la manera más flexible posible.

Los dos primeros puntos constituyen el núcleo del modelo relacional, ya que definen las características fundamentales que distinguen este modelo de los demás. Aunque los dos últimos puntos suelen considerarse como parte del modelo relacional, no son exclusivos de éste, y, de hecho, a menudo pueden ser usados muy ventajosamente en otros tipos de DBMS.

Ventajas

- Garantiza herramientas para evitar la duplicidad de registros, a través de campos claves o llaves.
- Garantiza la integridad referencial, así, al eliminar un registro elimina todos los registros relacionados dependientes.
- Favorece la normalización por ser más comprensible y aplicable.
- Compatibilidad y estandarización.
- Fiabilidad.
- Garantía de independencia de los datos.
- Existencia de numerosos sistemas comerciales entre los que se puede escoger y conseguir apoyo técnico.
- Conectividad garantizada con los lenguajes de programación estándar.

Desventajas

- Imposibilidad de representar conocimiento en forma de reglas.
- Inexistencia de mecanismos de herencia de propiedades (y por supuesto de métodos).
- Falta de poder expresivo (por ejemplo, para representar jerarquías).

Dificultad para gestionar datos no atómicos (por ejemplo, los valores estructurados de una estructura de rasgos).

Cardinalidad

Es el número de entidades con la cual otra entidad se puede asociar mediante una relación, considerando una relación entre las entidades A y B, la cardinalidad puede ser:

- Uno a uno - una instancia de la entidad A se relaciona con una y solamente una de la entidad B
- Uno a muchos o muchos a uno - cada instancia de la entidad A se relaciona con varias instancias de la entidad B.
- Muchos a muchos - cualquier instancia de la entidad A se relaciona con cualquier instancia de la entidad B.

Algebra Relacional

Se llama álgebra relacional a un conjunto de operaciones simples sobre tablas relacionales, a partir de las cuales se definen operaciones más complejas mediante composición. Definen, por tanto, un pequeño lenguaje de manipulación de datos.

El algebra relacional es un algebra en la cual:

- Sus operandos son relaciones (instancias) o variables que representan relaciones.

- Sus operadores están diseñados para hacer las tareas más comunes que se necesitan para manipular relaciones en una base de datos.
- El resultado es que el algebra relacional se puede utilizar como un lenguaje de consulta.

Operadores

- **Selección σ**

Selecciona un subconjunto de las tuplas de una relación. Tuplas seleccionadas son las que satisfacen cierto predicado lógico P. El predicado puede depender de los atributos de la relación y de valores constantes.

El operador σ toma una relación como argumento y el resultado es una nueva relación.

Podemos ver la selección como una operación que sirve para elegir algunas tuplas de una relación y eliminar el resto. Más concretamente, la selección es una operación que, a partir de una relación, obtiene una nueva relación formada por todas las tuplas de la relación de partida que cumplen una condición de selección especificada.

- **Proyección π**

Permite extraer columna de una relación, dando como resultado un **subconjunto vertical** de atributos de la relación, señalada con la letra griega pi mayúscula (π).

Como subíndice de π se coloca una lista de todos los atributos que se desea aparezcan en el resultado. La relación argumento se escribe después de π entre paréntesis.

- **Producto Cartesiano X**

El **producto cartesiano** de dos conjuntos **A x B** es el conjunto de todos los **pares ordenados** que se pueden formar con un elemento perteneciente al conjunto A y un elemento del conjunto B.

Sus elementos se colocan entre paréntesis, separados por coma.

- **Union \cup**

La unión es una operación que, a partir de dos relaciones, obtiene una nueva relación formada por todas las tuplas que están en alguna de las relaciones de partida.

La unión es una operación binaria, y la unión de dos relaciones T y S se indica $T \cup S$.

- **Interseccion \cap**

La intersección es una operación que, a partir de dos relaciones, obtiene una nueva relación formada por las tuplas que pertenecen a las dos relaciones de partida.

La intersección es una operación binaria; la intersección de dos relaciones T y S se indica $T \cap S$.

Normalización

La normalización es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más pequeñas, que además de ser más simples y más estables, son más fáciles de mantener. También se puede entender la normalización como una serie de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar un esquema que minimice los

problemas de lógica. Cada regla está basada en la que le antecede. La normalización se adoptó porque el viejo estilo de poner todos los datos en un solo lugar, como un archivo o una tabla de la base de datos, era ineficiente y conducía a errores de lógica cuando se trataban de manipular los datos.

La normalización también hace las cosas fáciles de entender. Los seres humanos tenemos la tendencia de simplificar las cosas al máximo. Lo hacemos con casi todo, desde los animales hasta con los automóviles. Vemos una imagen de gran tamaño y la hacemos más simple agrupando cosas similares juntas. Las guías que la normalización provee crean el marco de referencia para simplificar una estructura de datos compleja.

Otra ventaja de la normalización de base de datos es el consumo de espacio. Una base de datos normalizada ocupa menos espacio en disco que una no normalizada. Hay menos repetición de datos, lo que tiene como consecuencia un mucho menor uso de espacio en disco.

La normalización ayuda a clarificar la base de datos y a organizarla en partes más pequeñas y más fáciles de entender. En lugar de tener que entender una tabla gigantesca y monolítica que tiene muchos diferentes aspectos, sólo tenemos que entender los objetos pequeños y más tangibles, así como las relaciones que guardan con otros objetos también pequeños.

Primera Forma Normal 1FN

Una tabla se encuentra en primera forma normal si y solo si todas las columnas de todas las tablas contienen solo valores atómicos, es decir un único valor. Establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.

Segunda Forma Normal 2FN

La regla de la Segunda Forma Normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la llave primaria de la tabla para identificarlos.

Una vez alcanzado el nivel de la Segunda Forma Normal, se controlan la mayoría de los problemas de lógica. Podemos insertar un registro sin un exceso de datos en la mayoría de las tablas.

Tercera Forma Normal 3FN

Una tabla está normalizada en esta forma si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave.

Cuando las tablas están en la Tercera Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no deben haber datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir.

Cuarta Forma Normal 4FN

La 4NF se asegura de que las dependencias multivaluadas independientes estén correcta y eficientemente representadas en un diseño de base de datos. La 4NF

es el siguiente nivel de normalización después de la forma normal de Boyce-Codd (BCNF).

Una tabla está en 4NF si y solo si esta en Tercera forma normal o en BCNF (Cualquiera de ambas) y no posee dependencias multivaluadas no triviales. La definición de la 4NF confía en la noción de una dependencia multivaluada. Una tabla con una dependencia multivaluada es una donde la existencia de dos o más relaciones independientes muchos a muchos causa redundancia; y es esta redundancia la que es suprimida por la cuarta forma normal.

La forma normal de Boyce-Codd (BCNF) ocurre si una tabla está en tercera forma normal y además todo determinante es una clave candidata. En base de datos un atributo determinante es un atributo del que depende funcionalmente de manera completa algún otro atributo.

Quinta Forma Normal 5FN

La quinta forma normal se refiere a dependencias que son extrañas. Tiene que ver con tablas que pueden dividirse en subtablas, pero que no pueden reconstruirse. Quinta Forma Normal (5FN).

Una tabla esta en Quinta Forma Normal (5FN) o Forma Normal de Proyección-Unión si está en 4FN y las únicas dependencias que existen son las dependencias de unión de una tabla con sus proyecciones relacionándose entre las distintas proyecciones mediante la clave primaria o cualquier clave alternativa. La 5FN se emplea cuando en una misma tabla tenemos mucha información redundante, con pocos atributos o cuando una tabla posee una gran cantidad de atributos y se hace por ello inmanejable.

Para conseguir que una tabla 4FN con gran cantidad de atributos esté en 5FN, se parte la tabla original en tantas tablas como se desee, teniendo cada una de ellas en común con las demás los campos que forman la clave primaria en la tabla original.

2.5 Características, ventajas y desventajas del motor de Bases de Datos Relacionales PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

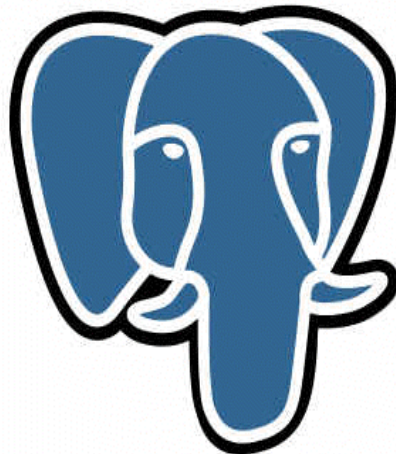
Sus características técnicas la hacen una de las bases de datos más potentes y robustas del mercado. Su desarrollo comenzó hace más de 16 años, y durante este tiempo, estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema.

Características

- Está desarrollado en C, con herramientas como Yacc y Lex.
- La API de acceso al SGBD se encuentra disponible en C, C++, Java, Perl, PHP, Python y TCL, entre otros.
- Cuenta con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario.
- Su administración se basa en usuarios y privilegios.
- Sus opciones de conectividad abarcan TCP/IP, sockets Unix y sockets NT, además de soportar completamente ODBC.

- Los mensajes de error pueden estar en español y hacer ordenaciones correctas con palabras acentuadas o con la letra 'ñ'.
- Es altamente confiable en cuanto a estabilidad se refiere.
- Puede extenderse con librerías externas para soportar encriptación, búsquedas por similitud fonética (soundex), etc.
- Control de concurrencia multiversión, lo que mejora sensiblemente las operaciones de bloqueo y transacciones en sistemas multiusuario.
- Soporte para vistas, claves foráneas, integridad referencial, disparadores, procedimientos almacenados, subconsultas y casi todos los tipos y operadores soportados en SQL92 y SQL99.

PostgreSQL



Ventajas

- A pesar de que la velocidad de respuesta pueda parecer deficiente en bases de datos pequeñas, esa velocidad se mantiene al aumentar el tamaño de la base de datos, cosa que no sucede con otros programas, que se ralentizan brutalmente.
- Instalación ilimitada ya que no hay costo asociado a la licencia de *software*. Esto permite un negocio más rentable con instalaciones a gran escala, no

debemos preocuparnos por ser auditados para ver si cumplimos con la licencia, y hay flexibilidad y desarrollo sin costos adicionales de licenciamiento.

- Estabilidad y Confiabilidad Legendarias, es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad. Ni una sola vez. Simplemente funciona.
- Extensible: El código fuente está disponible de forma gratuita, para que quien necesite extender o personalizar el programa pueda hacerlo sin costes.
- Multiplataforma: Está disponible en casi cualquier *Unix*, con 34 plataformas en la última versión estable, además de una versión nativa de *Windows* en estado de prueba.
- Diseñado para ambientes de alto volumen: Utilizando una estrategia de almacenamiento de filas llamada MVCC, consigue mejor respuesta en grandes volúmenes. Además, MVCC permite a los accesos de solo lectura continuar leyendo datos consistentes durante la actualización de registros. Los principales proveedores de sistemas de bases de datos comerciales usan también esta tecnología, por las mismas razones.
- Herramientas gráficas de diseño y administración de bases de datos. Existen varias herramientas gráficas de alta calidad para administrar las bases de datos (pgAdmin , pgAccess) y para hacer diseño de bases de datos (Tora , Data Architect).
- Soporta los tipos de datos, cláusulas, funciones y comandos de tipo estándar *SQL92/SQL99* y extendidos propios de *PostgreSQL*.
- Puede operar sobre distintas plataformas, incluyendo *Linux*, *Windows*, *Unix*, *Solaris* y *MacOS X*.
- Buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.
- Gran capacidad de almacenamiento.

- Buena escalabilidad ya que es capaz de ajustarse al número de CPU y a la cantidad de memoria disponible de forma óptima, soportando una mayor cantidad de peticiones simultáneas a la base de datos de forma correcta.

Desventajas

- En comparación con *MySQL* es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección que no tiene *MySQL*.
- Soporte en línea: Hay foros oficiales, pero no hay una ayuda obligatoria.
- Consume más recursos que *MySQL*.
- La sintaxis de algunos de sus comandos o sentencias no es nada intuitiva.
- Sin experticia, configurar llega a ser un caos.
- Es fácil de vulnerar sin protección adecuada.
- El motor MyISAM es instalado por defecto y carece de capacidades de integridad relacional.
- InnoDB genera mucho footprint en memoria al indizar.
- El toolset empresarial tiene un costo adicional por suscripción anual.
- Realizar revisiones llega a ser una labor manual y tediosa para el DBA.

CAPITULO III. ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA

CAPITULO III. ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA

3.1 Análisis del problema

La figura 3.1.1 presenta el diagrama general del proceso que se lleva en la actualidad para llevar a cabo la evaluación a los profesores a través de las encuestas repartidas a los alumnos, partiendo desde la selección del comité encargado de la organización y evaluación hasta los acuerdos que se tendrán con el profesor de acuerdo a los resultados, para determinar si requieren capacitación o simplemente reconocerles.

A continuación iremos describiendo cada bloque a fin de clarificar mejor las actividades y los pasos involucrados durante la evaluación.

- **Se designa comité organizador:** deberá estar compuesto por personas imparciales en el proceso, asegurando la transparencia. Generalmente designados por el área de Pedagogía.
- **Propuesta de preguntas para evaluación:** se reúne un comité para decidir las preguntas que sean relevantes y que estas puedan reflejar el desempeño del profesor, además de considerar las propuestas de años anteriores.
- **Se calendariza visita a los grupos → Se visita al grupo en la fecha → Se interrumpe la clase:** Se presenta la agenda con los días y horarios programados para asistir a cada uno de los grupos a la aplicación del examen.

El número de personas que componen al comité es muy reducido, por lo que el periodo para abarcar a todos los grupos tiende a ser muy prolongado.

No siempre se cumple con el día ni horario ya que puede que un grupo tarde más en contestar la evaluación, también implica la disposición del profesor ya que a veces no permite la entrada a la hora programada haciéndolos esperar y atrasándolos con los demás grupos a visitar.

Al interrumpirse la clase se pierde el ritmo que se tenía, provocando que los alumnos pierdan interés en ella después de la evaluación.

Al interrumpirse la clase los alumnos no muestran interés en la evaluación por estar viendo un tema importante y esto provoca que se conteste de manera indiferente.

- **Se entrega la encuesta al alumno → El alumno llena la encuesta:**

El comité imparte las instrucciones necesarias a los alumnos para el correcto llenado en la evaluación.

Cabe mencionar que la encuesta es entregada físicamente en papel por lo que el uso del material es considerado como un gasto.

Puede presentarse el mal llenado de la encuesta por parte del alumno, lo cual impediría vaciar al 100 las respuestas en los indicadores.

- **Recolección de encuesta por el comité:** Una vez finalizada la encuesta esta es recolectada por los integrantes del comité, hacen el conteo de evaluaciones entregadas al inicio y evaluaciones recibidas.

- **Vaciado de respuestas → ¿Respuesta ilegible? → Se captura para conteo**

Las respuestas de las evaluaciones deben ser capturadas a una hoja de cálculo para su concentración y análisis.

Las evaluaciones en papel pueden contener correcciones, tachaduras y borrones por lo que no siempre el 100 % de las respuestas de los

alumnos es legible, teniendo que anular algunas respuestas provocando pérdida de información.

- **Vaciado de respuestas en indicadores → Actualización de los diferentes reportes**

Una vez que se tienen reunidas todas las respuestas de los alumnos en medio digitales, se concentra la información en los diversos layout para poder obtener los diferentes indicadores que servirán para la evaluación al profesor.

- **Vo Bo. comité → Retroalimentación a profesores →**

- **Acuerdos para capacitación con el profesor**

El comité valida que los indicadores sean los mejores para la evaluación al profesor se tiene una sesión de retroalimentación acerca de sus resultados se le reconoce y también se hacen acuerdos sobre las áreas de oportunidad en las que debe mejorar sugiriéndole, cursos talleres o capacitaciones.

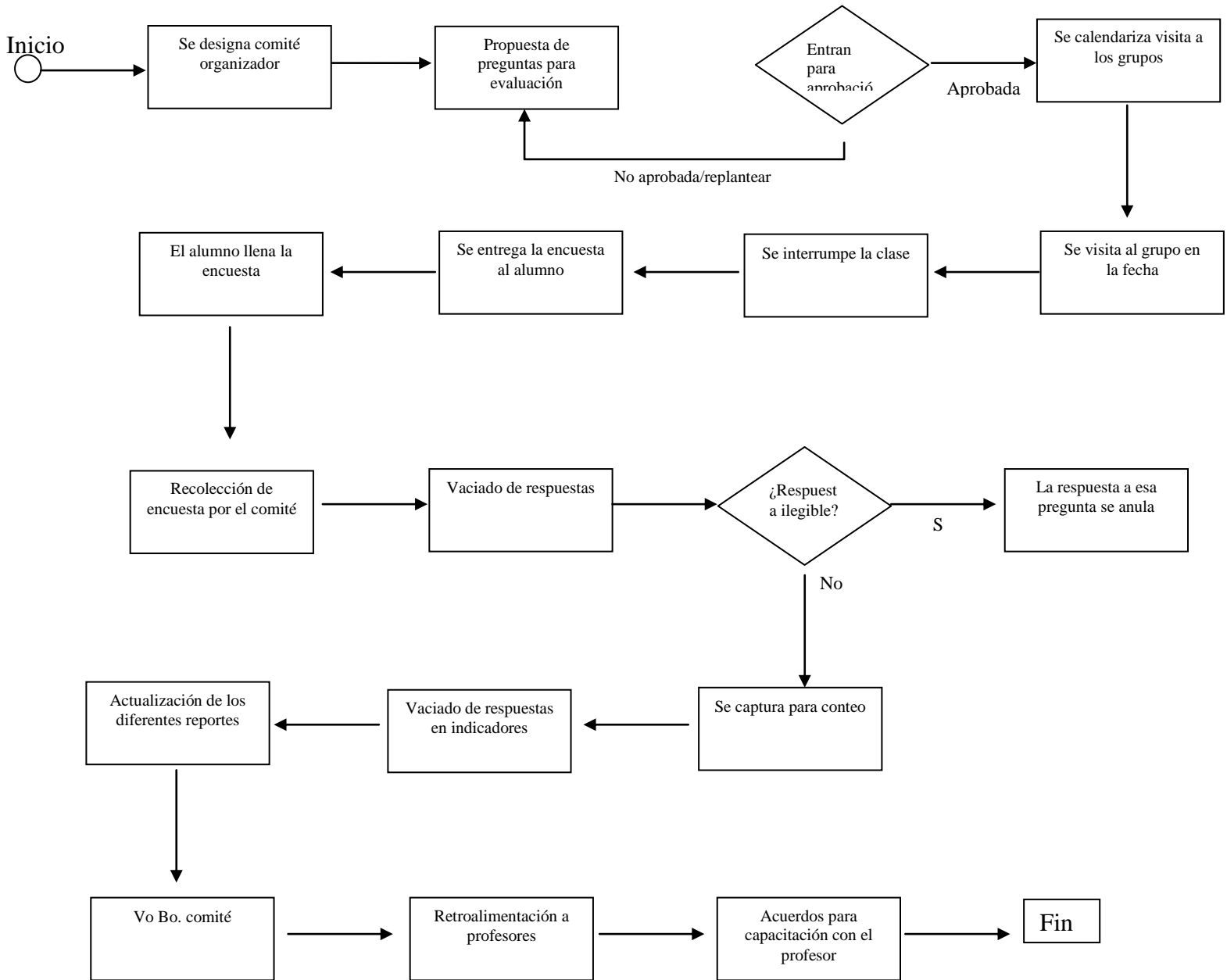


Figura 3.1.1 Diagrama Proceso de Evaluación

3.2 Recopilación y análisis de la información

La evaluación es una actividad reflexiva que nos permite conocer la calidad de los procesos y los logros alcanzados en el desarrollo del proyecto. La evaluación es una valoración sistemática que facilita el conocimiento minucioso de los procesos aplicados y, fundamentalmente las decisiones futuras de cambio que nos proponemos llevar a cabo.

La evaluación del desempeño profesional del docente es un proceso sistemático de obtención de datos válidos y fiables, con el objetivo de comprobar y valorar el efecto educativo que produce en los alumnos el despliegue de sus capacidades pedagógicas, su emocionalidad, responsabilidad laboral y la naturaleza de sus relaciones interpersonales con alumnos, padres, directivos, colegas y representantes de las instituciones de la comunidad.

Evaluar al profesorado no es proyectar en él las deficiencias o razonables limitaciones del Sistema Educativo, sino es asumir un nuevo estilo, clima y horizonte de reflexión compartida para optimizar y posibilitar espacios reales de desarrollo profesional de los docentes, de generación de culturas innovadoras en los centros educativos.

La evaluación es un juicio de valor que necesita referentes bien consolidados a los que tender y con los que contrastar la realidad evaluada, más esta constatación exigiría plena coincidencia en la identificación de tales referentes y en su aplicación.

Toda actividad de evaluación es un proceso en tres etapas:

- Levantamiento de información, que puede ser o no instrumentada.
- Análisis de esta información y juicio sobre el resultado de ese análisis.

- Toma de decisiones de acuerdo con el juicio emitido.

De esta definición no se infiere directamente que la evaluación se tenga que identificar con examen y que deba implicar necesariamente un acto administrativo. Esta identificación, que es muy frecuente en el ámbito escolar, es el resultado de una visión parcial de la función que tiene la evaluación en el proceso de enseñanza - aprendizaje.

La evaluación de los aprendizajes presenta básicamente dos funciones:

- **Una de carácter social de selección y de clasificación**, pero también de orientación del alumnado.
- **Una de carácter pedagógico, de regulación del proceso de enseñanza - aprendizaje**, es decir, de reconocimiento de los cambios que se han de introducir progresivamente en este proceso para que todos los alumnos aprendan en forma significativa.

La primera de estas funciones pretende, esencialmente, informar de la progresión de sus aprendizajes al alumno y a sus padres, y determinar qué alumnos han adquirido los conocimientos necesarios para poder acreditarles la certificación correspondiente que la sociedad requiere del sistema escolar. Por lo tanto, esta función es de carácter social, pues constata y/o certifica la adquisición de unos conocimientos al terminar una unidad de trabajo, se inserta necesariamente al final de un período de formación del que se quiere hacer un balance o al final de un curso o ciclo.

La segunda de dichas funciones es de carácter pedagógico o formativo, pues aporta información útil para la adaptación de las actividades de enseñanza - aprendizaje a las necesidades del alumnado y de este modo mejorar la cualidad

de la enseñanza en general. Se inserta en el proceso de formación, ya sea en su inicio, durante él o al final, pero siempre con la finalidad de mejorar el aprendizaje cuando aún se está a tiempo.

Fases de la evaluación

La evaluación se caracteriza por ser sistemática y es, precisamente, en este apartado donde detallamos más el asunto. Toda evaluación auténtica y responsable debe preverse desde el momento mismo de la programación de aula, cuando el profesor establece los indicadores para cada capacidad y actitud.

El proceso de evaluación comprende las siguientes etapas:

- **Planificación de la evaluación**

Planificar la evaluación implica esencialmente dar respuesta a las siguientes interrogantes: qué, para qué, cómo, cuándo se evaluará y con qué instrumentos. De este modo, la evaluación se convierte en un acto pensado y ejecutado intencional mente, minimizando la improvisación.

Intentamos dar respuesta a cada una de estas preguntas:

¿Qué evaluaré?

Se trata de seleccionar qué capacidades y qué actitudes evaluaremos durante una unidad didáctica o sesión de aprendizaje, en función de las intenciones de enseñanza.

¿Para qué evaluaré?

Precisamos para qué nos servirá la información que recojamos: para detectar el estado inicial de los estudiantes, para regular el proceso, para determinar el nivel de desarrollo alcanzado en alguna capacidad, etc.

¿Cómo evaluaré?

Seleccionamos las técnicas y procedimientos más adecuados para evaluar las capacidades, conocimientos y actitudes, considerando además los propósitos que se persigue al evaluar.

¿Con qué instrumentos?

Seleccionamos e indicamos los instrumentos más adecuados para evaluar los aprendizajes. Los indicadores de evaluación son un referente importante para optar por uno u otro instrumento.

¿Cuándo evaluaré?

Precisamos el momento en que se realizará la aplicación de los instrumentos. Esto no quita que se pueda recoger información en cualquier momento, a partir de actividades no programadas o como parte de la evaluación no formal.

- **Recopilación y selección de información**

La obtención de información sobre los aprendizajes de los estudiantes, se realiza mediante técnicas formales, semiformales o no formales. De toda la información obtenida se deberá seleccionar la que resulte más confiable y significativa.

La información es más confiable cuando procede de la aplicación sistemática de técnicas e instrumentos y no del simple azar. Serán preferibles, por ejemplo, los

datos provenientes de una lista de cotejo antes que los derivados de una observación improvisada. Por otra parte, la información es significativa si se refiere a aspectos relevantes de los aprendizajes.

- **Interpretación y valoración de la información**

Se realiza en términos del grado de desarrollo de los aprendizajes establecidos en cada área. Se trata de encontrar sentido a los resultados de la evaluación, determinar si son coherentes o no con los propósitos planteados (y sobre todo con los rendimientos anteriores de los estudiantes) y emitir un juicio de valor.

En la interpretación de los resultados también se considera las reales posibilidades de los alumnos, sus ritmos de aprendizaje, la regularidad demostrada, etc., porque ello determina el mayor o menor desarrollo de las capacidades y actitudes. Esta es la base para una valoración justa de los resultados.

Valoramos los resultados cuando les otorgamos algún código representativo que comunica lo que el alumno fue capaz de realizar. Hay diferentes escalas de valoración: numéricas, literales o gráficas. Pero, también se puede emplear un estilo descriptivo del estado en que se encuentra el aprendizaje de los estudiantes.

- **Comunicación de los resultados**

Esto significa que se analiza y se dialoga acerca del proceso educativo con la participación de los alumnos, docentes, de tal manera que los resultados de la evaluación son conocidos por todos los interesados. Así, todos se involucran en el proceso y los resultados son más significativos, permitiendo tomar decisiones oportunas y pertinentes.

Los instrumentos empleados para la comunicación de los resultados son los registros auxiliares del docente, los registros consolidados de evaluación y las libretas de información al padre de familia. Sin embargo, la comunicación de los resultados también se puede hacer en forma personal, mediante informes orales o conversaciones sobre la situación de los aprendizajes de los estudiantes.

- **Toma de decisiones**

Los resultados de la evaluación deben llevarnos a aplicar medidas pertinentes y oportunas para mejorar el proceso de aprendizaje. Esto implica volver sobre lo actuado para atender aquellos aspectos que requieran reformulaciones, profundización, refuerzo o recuperación. Las deficiencias que se produzcan pueden deberse tanto a las estrategias empleadas por el docente como a la propia evaluación. Cuando sucede esto último es necesario que reflexionemos sobre la misma evaluación, para corroborar si realmente existe coherencia entre los aprendizajes previstos y lo que realmente se ha evaluado.

Para una adecuada toma de decisiones, se debe realizar un análisis de los resultados obtenidos. Las intencionalidades de las áreas curriculares que convergen en el logro de una formación integral de los educandos; en consecuencia, los procesos de enseñanza, aprendizaje y evaluación se deben orientar en ese sentido. El proceso formativo abarca, por consiguiente, todas las dimensiones del desarrollo del estudiante.

El modelo de evaluación del aprendizaje asume los conceptos y enfoques vertidos anteriormente, y se describe como un proceso sistémico cuyo punto de partida es determinar la situación en que se encuentran los estudiantes respecto a las intencionalidades del currículo. A partir de estos datos, el docente proporciona actividades de apoyo para que los alumnos desarrollen las capacidades y actitudes, y adquieran los conocimientos previstos. Durante el desarrollo de las actividades, el docente, mediante un conjunto de procedimientos formales,

semiformales o no formales, recoge información sobre el proceso de aprendizaje y enseñanza con la finalidad de regularlos, mediante mecanismos de realimentación. Al finalizar un período determinado, es necesario tener información sobre el desempeño de los estudiantes respecto de los aprendizajes esperados.

Análisis de información

La gestión pedagógica docente comprende una diversidad de acciones que están enmarcados en las siguientes dimensiones:

- Dominio tecnológico: Adecuación del Diseño Curricular, a la planificación curricular, la ejecución de la programación curricular, la aplicación metodológica y uso de medios y materiales didácticos.
- Dominio científico: Formación profesional de conocimientos sobre los contenidos de las áreas académicas y la forma como se trasmite su conocimiento a los estudiantes.
- Responsabilidad en el desempeño de sus funciones: Es puntual y cumple con sus funciones.
- Relaciones interpersonales: Relación del docente con los estudiantes y con los demás integrantes de la comunidad educativa.
- Formación en valores éticos.

Importancia y alcances de la investigación

En la recopilación y análisis de la información se justifica por los siguientes aspectos: Pedagógicos: Muchas investigaciones que se analizan en el contexto de este trabajo, definen la importancia de indagar las características del Desempeño

Docente y el Rendimiento Escolar en instituciones educativas a sabiendas de la incidencia que este aspecto puede generar en el desempeño y productividad de sus miembros. De allí que pueden generarse alternativas que pudieren contribuir al mejoramiento continuo de la educación en cualquier nivel y modalidad del sistema educativo.

- **Técnicos:** Es necesario e importante que el docente aporte herramientas, originadas de estudios sistemáticos y con el rigor científico necesario, para recuperar niveles óptimos de rendimiento académicos educativos de los estudiantes donde ejercen su labor.
- **Prácticos:** Los resultados de esta investigación nos permitirán conocer el estado en que se encuentra la relación de las variables de Desempeño Docente y el Rendimiento Académico. La teoría sistematizada de ambas variables, nos permitirá formular recomendaciones a las autoridades de las instituciones educativas.

Los resultados del presente trabajo de tesis podrían servir para que la institución educativa tome las decisiones correctivas para mejorar la calidad educativa.

Por otro lado, el presente trabajo tiene como alcance a los investigadores en el campo educativo, docentes y autoridades educativas, considerando que la investigación debe practicarse para solucionar los diferentes problemas que influyen en el proceso de enseñanza-aprendizaje de los estudiantes.

Metodología

Hipótesis general: Existe relación significativa entre la evaluación del desempeño docente y el rendimiento académico de los estudiantes

VARIABLES:

Variable independiente: Evaluación del desempeño docente

Variable dependiente: Rendimiento académico de los estudiantes. Vea tabla 3.2.1, 3.2.2, 3.2.3 y 3.2.4

CUADRO DE DIMENSIONES DE VARIABLES E INDICADORES				
VARIABLES	DIMENSIONES	INDICADORES	INFORMANTE	INSTRUMENTOS
VARIABLE Desempeño Docente	DOMINIO TECNOLÓGICO (Metodología de Enseñanza, Medios y Materiales Didácticos y Evaluación)	<ul style="list-style-type: none"> • Presentación y exposición • Desarrollo de los contenidos del área académica • Preparación de clases. • Motivación para la participación. • Pertinencia en trabajos extra – clase e investigación. • Desarrollo de habilidades del pensamiento. • Congruencia entre objetivos y contenidos. • Utilización de recursos didácticos. • Diversificación de métodos de enseñanza. • Uso de formas de evaluación adecuadas y consistentes. • Relación entre la evaluación y los contenidos. • Identificación de conocimientos y fomento de la auto evaluación. • Criterios de evaluación. • Retroalimentación sobre aciertos y errores de la evaluación • Elaboración de materiales didácticos. • Diseño de estrategias de aprendizaje. • Apoyo de actividades extracurriculares. • Propuesta de acciones de apoyo a la docencia. • Participación de revisiones curriculares, diseño y elaboración. 	Estudiantes	Cuestionario

Tabla 3.2.1 Dominio Tecnológico

	<p style="text-align: center;">DOMINIO CIENTIFICO (Rasgos Profesionales y Académicos)</p>	<ul style="list-style-type: none"> • Dominio y actualización de conocimientos. • Empleo de términos técnicos de la especialidad. • Claridad y solvencia en la exposición de temas. • Dominio del tema. • Destreza en la solución de problemas. • Actividades que facilita la comprensión de la signatura. • Uso de bibliografía actualizada y relevante. • Relación de la asignatura con otras asignaturas. • Presentación de ejemplos de sus investigaciones y experiencias. • Acciones de docencia e investigación. • Participación de proyectos. • Conocimiento del área de desempeño. • Emplea y recomienda el uso de bibliografía actualizada. • Participación en formación y actualización profesional. • Interés e iniciativa por el desarrollo académico de la institución. • Participación en ponencias. 	Estudiantes	Cuestionario
--	--	---	--------------------	---------------------

Tabla 3.2.2 Dominio Científico

CUADRO DE DIMENSIONES DE VARIABLES E INDICADORES				
Variables	Dimensiones	Indicadores	Informante	Instrumentos
VARIABLE Desempeño Docente	RELACIONES INTERPERSONALES (interrelación con los estudiantes y los demás miembros de la Institución Educativa)	<ul style="list-style-type: none"> • Respeto y dignidad en su trato con los demás. • Recepción de sugerencias. • Propicio de ambiente para el aprendizaje. • Proyección de actitudes a la Institución Educativa, la vida y la sociedad. • Trato imparcial. • Disciplina. • Contribución a la formación integral y humana. • Atención de reclamos a la forma de calificación en las evaluaciones. • Apoyo a los miembros de la comunidad. • Colaboración con un buen ambiente de trabajo. • Actitud frente a los cambios. • Comportamiento profesional. • Relaciones de respeto y cordialidad con los alumnos. • Respeto a las ideas de los demás. • Usa de conductos regulares en su institución. • Proyecta imagen como modelo personal hacia los alumnos. 	Estudiantes	Cuestionario
	FORMACIÓN EN VALORES ÉTICOS	<ul style="list-style-type: none"> • Actitudes, valores y principios universales y profesionales. • Interés por la asignatura, la participación e investigación. • Fomento del uso de la expresión oral y escrita. • Reflexión sobre que, el porqué y como lo haces. 		

Tabla 3.2.3 Relaciones interpersonales y valores éticos

La variable desempeño docente en la presente investigación asume cinco niveles: Excelente, bueno, regular, deficiente y muy deficiente.

- Rendimiento académico excelente
- Rendimiento académico bueno
- Rendimiento académico regular
- Rendimiento académico deficiente
- Rendimiento académico muy deficiente

VARIABLE	DIMENSIONES	INDICADORES
Rendimiento Académico	Manejo de Información	<ul style="list-style-type: none"> • Identifica • Discrimina • Analiza/Selecciona • Organiza • Infiere • Interpreta/Juzga • Comunica • Evalúa
	Comprensión Espacio-Temporal	<ul style="list-style-type: none"> • Utiliza/Crea • Predice/Formula • Identifica • Analiza/Interpreta. • Juzga/Evalúa • Argumenta/Comunica
	Juicio Critico	<ul style="list-style-type: none"> • Identifica/Analiza • Utiliza • Imagina/Propone • Argumenta • Asume • Planifica/Selecciona • Formula/Juzga

Tabla 3.2.4 Rendimiento Académico de los estudiantes

La variable rendimiento académico en la presente evaluación asume cinco niveles: Excelente, bueno, regular, deficiente y muy deficiente.

- Rendimiento académico excelente
- Rendimiento académico bueno
- Rendimiento académico regular
- Rendimiento académico deficiente
- Rendimiento académico muy deficiente

Método de la investigación

La presente evaluación es de tipo descriptivo-correlacional, no experimental, porque estudia las características del desempeño docente y su relación con el rendimiento académico de los estudiantes.

Diseño de la investigación

La presente evaluación se centrará en el enfoque cuantitativo-cualitativo, en un estudio de recolección de datos en un solo momento, en un tiempo único.

Población de Estudio

La población de estudio para la presente evaluación docente está constituida por los alumnos de la Institución educativa

Selección y validación de los instrumentos

Para la ejecución del proyecto de investigación se utilizó un tipo de instrumento de recolección de datos: el cuestionario (encuestas) aplicado a los estudiantes para recoger información referente a las dimensiones de la variable desempeño docente.

Validez de los instrumentos

Para su validación se realizó en base al marco teórico de la categoría de “validez de contenido”, utilizando el procedimiento de juicio de expertos calificados que determinaron la adecuación de los ítems de los respectivos instrumentos.

Descripción de recolección de datos

Para llevar a cabo la aplicación de los instrumentos de recolección de información se procedió a coordinar con los directores, profesores del área académica.

La encuesta se aplicará a los estudiantes para recoger información referente a las dimensiones de la variable desempeño docente.

A continuación se muestra la encuesta de evaluación docente que se realiza actualmente en la Institución educativa.

ENCUESTA DE EVALUACIÓN PARA MEDIR EL DESEMPEÑO DOCENTE

Estimado Estudiante:

La presente encuesta es para un trabajo de investigación, relacionado con el desempeño de tu profesor o profesora del área académica. Agradecemos anticipadamente tu valiosa colaboración llenando de manera sincera y responsable la siguiente encuesta.

INSTRUCCIONES: La información que nos proporcionarás será solo de conocimiento del investigador, por tanto evalúa a tus docentes en forma objetiva y veraz respondiendo a las interrogantes, en función a la siguiente escala: Vea Formato 3.2.5.

1 = Nunca	2 = Algunas veces	3 = Frecuentemente	4 = Casi siempre	5 = Siempre
-----------	-------------------	--------------------	------------------	-------------

N°	DIMENSIONES	ITEMS RELACIONADOS CON EL DOCENTE	ESCALA				
			1	2	3	4	5
1	Dominio Tecnológico	El docente presenta y expone con claridad (propósito, objetivo, contenido, criterios de evaluación y bibliografía) al inicio de clases.					
2		Desarrolla contenidos del Area Académica					
3		El docente prepara sus clases con anticipación.					
4		Motiva permanentemente la participación de los alumnos con preguntas, debates y trabajo en equipo.					
5		El docente nos deja tarea de lo que vimos en clase y promueve la investigación.					
6		Promueve habilidades del pensamiento (análisis, creatividad, valoración, crítica y solución de problemas)					
7		El desarrollo del área académica te permite desarrollar las capacidades del area.					
8		El docente utiliza adecuadamente diferentes materiales para facilitar el proceso de enseñanza-aprendizaje de los estudiantes.					
9		El docente utiliza diferentes métodos de enseñanza de acuerdo con el tema de la sesión de aprendizaje.					
10		El docente utiliza formas de evaluación adecuadas con lo establecido al inicio del área académica					
11		El docente aplica evaluaciones que se relacionan con el contenido del área académica					
12		El docente identifica los conocimientos previos de los alumnos y fomenta la autoevaluación.					
13		El docente es justo en las calificaciones que me asigna.					
14		El docente refuerza los aprendizajes adecuadamente de los estudiantes sobre aciertos y errores de la evaluación.					
15	Dominio Científico	El docente muestra un sólido dominio y actualización de conocimientos del área académica que enseña.					
16		El docente explica con propiedad los términos técnicos del área académica					
17		El docente se expresa con claridad y dominio en la exposición de los temas del área académica a los estudiantes.					
18		El docente responde a las preguntas, dudas o inquietudes de los estudiantes en forma clara y precisa.					

N°	DIMENSIONES	ITEMS RELACIONADOS CON EL DOCENTE	ESCALA				
			1	2	3	4	5
19	Dominio Científico	El docente realiza actividades y/o visitas de estudio que facilita la comprensión del área académica de los estudiantes.					
20		El docente emplea y recomienda el uso de bibliografía actualizada y relevante.					
21		El docente relaciona el área académica con los contenidos de otras áreas académicas.					
22		Es exigente con sus alumnos y solicita calidad en todas las actividades de enseñanza-aprendizaje.					
23	Responsabilidad	El docente asiste puntualmente y con regularidad a sus clases					
24		El docente entrega a tiempo los documentos educativos (sílabos, separatas, bibliografía, etc.) a los estudiantes.					
25		El docente informa oportunamente los resultados de la evaluación a los estudiantes.					
26		El docente inicia y concluye el desarrollo de los contenidos de su área académica dentro del tiempo programado.					
27	Relaciones Interpersonales	El docente es respetuoso, acepta ideas diferentes a las suyas y muestra buen trato hacia los estudiantes y los demás integrantes de la comunidad educativa.					
28		El docente acepta las sugerencias de los estudiantes para el desarrollo y la evaluación del área académica.					
29		El docente propicia un ambiente favorable de comunicación para el aprendizaje.					
30		El docente proyecta actitudes positivas hacia la I.E, la vida y la responsabilidad social.					
31		El docente muestra un trato justo con todos los estudiantes.					
32		El docente se hace respetar y mantiene la debida disciplina en la institución educativa.					
33		El docente contribuye a la formación integral de los estudiantes.					
34		El docente atiende a los reclamos de los estudiantes en relación a la forma en que califica las evaluaciones.					
35	Formación en Valores Éticos	El docente muestra actitudes, valores y principios universales éticos.					
36		El docente promueve y estimula la práctica de valores ético - morales en clases en los estudiantes.					
37		Fomenta el uso correcto de la expresión oral y escrita.					
38		Logra que tomes conciencia de lo que haces, el ¿Por qué? y ¿Cómo lo haces?					

Formato 3.2.5 Encuesta

3.3 Requerimientos generales y específicos de la aplicación

Por definición los requerimientos en aplicaciones de software, son las características que debe tener el software instalado en una computadora para poder soportar y/o ejecutar una aplicación o un dispositivo específico. Es una descripción de algo que el sistema es capaz de hacer con el objeto de satisfacer el propósito del sistema. Es decir, los requerimientos son lo que los clientes/usuarios esperan que haga el sistema.

Como objetivo principal tenemos el poder crear un sistema en línea que pueda ser accesible desde cualquier computadora, independientemente del sistema operativo que utilice; con lo cual se les facilitará a los alumnos el proceso de evaluación de sus profesores.

Requerimientos Generales

- El sistema al estar enfocado a alumnos deberá de contar con una interfaz amigable y con un diseño moderno donde la información esté muy bien organizada, lo cual motive al usuario a realizar su evaluación.
- Las pantallas deberán de tener un estilo en común, es decir, la posición de los botones y las preguntas deben de ser similares, así como el fondo y los colores.
- Los botones deberán estar centrados y organizados tomando como referencia los cuadros de texto que se utilicen para que visualmente no se vea tan saturado.
- Los textos que contengan información importante deberán estar en negritas y con mayúsculas, con un tamaño un poco más grande que las demás.

- Las imágenes o logos que se utilizarán deberán adecuarse a las pantallas de los usuarios, evitando que se distorsionen y no se puedan ver claramente.
- El sistema será desarrollado utilizando lenguajes de programación orientados a web, específicamente:
 - HTML: con el cual manejaremos la estructura de la página.
 - PHP: que es un lenguaje de programación de uso general de script del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.
 - Javascript: el cual puede afectar el comportamiento de navegadores web.
- El Sistema de Evaluación deberá operar en equipos que cuenten como mínimo con las siguientes especificaciones:
 - Servidor: 900 Mhz, RAM 512 MB, 200 MB de espacio en disco para la instalación
 - Cliente: Pentium 233 MHz (Recomendado: Pentium 500MHz o superior), 64 MB RAM (Recomendado: 128 MB RAM o más), 52 MB de espacio en disco y monitor con resolución de 1024 x 720.
- Todos los datos serán almacenados en una base de datos utilizando el manejador de base de datos Postgresql, ya que es rápido, gratuito y se acopla al sistema que se desarrollará.
- Se deberá contar con una metodología de respaldos de las tablas a utilizar, sobre todo donde se guardan la información de la evaluación. También un plan de mantenimiento de la base de datos semestralmente, haciendo

backup de las tablas necesarias y limpiando las que se utilizaran el siguiente semestre.

- El servidor deberá de estar instalado en algunas de las computadoras con que cuente la institución, y este dentro del mismo edificio o Centro de Cómputo, si es que cuentan con uno.
- El administrador del sistema será el único en poder acceder a toda la información almacenada en la base de datos, así como la realización de los reportes correspondientes a entregar.

Requerimientos de Seguridad

- El sistema de evaluación docente deberá de tener un control de Login, ya que es el control de inicio de sesión más simple y soporta las situaciones más comunes; identificarse utilizando un nombre de usuario y una contraseña. El control incluye los cuadros de texto usuario y contraseña y una casilla de verificación para los usuarios que deseen comprometer la seguridad de la contraseña almacenándola en la máquina.

Función de autenticación de usuario

- Introducción: Mediante la introducción de su número de cuenta o id y contraseña, los usuarios registrados y el administrador accederán a la aplicación y a sus características asociadas.
- Entradas: Para el campo de número de cuenta o id se requieren solo números. Para el campo de contraseña de requieren alfanuméricos.
- Proceso: Se realizará una comprobación de que el usuario y su contraseña coincidan dentro de la base de datos, si su identificación es correcta se

mostrará la pantalla de instrucciones. En caso contrario aparecerá una pantalla con un aviso de que sus datos no son correctos.

- Salida: En caso de que la identificación sea correcta nos llevará a una pantalla con las instrucciones necesarias para iniciar la evaluación y visualización de su tira de materias.

Funciones de visualización del cuestionario de evaluación

- Introducción: Se selecciona al profesor a evaluar.
- Proceso: El sistema cargará las preguntas por partes y el usuario deberá responder todas, sino no lo hace no podrá avanzar a la siguiente parte del cuestionario.
- Salida: En la pantalla aparecerá que su registro fue guardado satisfactoriamente y regresará al usuario para que continúe calificando a los demás profesores. Al concluir con todos se le generará un número de folio, con lo cual se verifica que se terminó por completo la evaluación.

Requisitos Particulares

- El Sistema de Evaluación tendrá un modulo de generación de reportes. El modulo no estará integrado al sistema, será desarrollado aparte y en otro lenguaje, será una aplicación de escritorio la cual será entregada al administrador, quien es el que generará los reportes cada semestre.
- El módulo de Generación de Reportes será desarrollado en Java, como aplicación de escritorio, tendrá acceso a la base de datos donde se encuentran los resultados de la evaluación, listos para ser procesados.

- Tendrá una interfaz estilo Windows, con menús para escoger qué reporte se quiere generar. Se necesitará tener instalado JVM en el equipo para poder visualizar la aplicación.
- Tendrá dos tipos de informe individual. El informe individual presenta de manera resumida los resultados para cada profesor en cada uno de los grupos en que imparte clase, según el número de factores que se hayan obtenido.
- Para emitir los informes correspondientes a cada profesor por materia y grupo, se les ha asignado una clave de identificación única, de igual manera, los grupos y materias tienen su propia clave de identificación.
- Se entregaran dos informes, el informe individual por factores o por escala y el informe por reactivos. Los informes se emiten por profesor-asignatura-grupo.
- El informe individual por factores o por escala deberá contener el número y nombre de cada uno de los factores que se obtuvieron en el análisis, y cuando en el proceso están definidas escalas fijas de evaluación. En él se presenta el número de alumnos que respondieron completamente el cuestionario, la media y desviación estándar de cada factor (o escala) así como los valores mínimos y máximos y la definición de cada factor (o escala) según los reactivos que lo conforman.
- Para interpretar los datos, el informe deberá mostrar lo siguiente:

Casi nunca	(0.00-0.99)
Algunas veces	(1.00-1.99)
Frecuentemente	(2.00-2.99)
Casi Siempre	(3.00-3.99)

Siempre

(4.00-5.00)

- En el informe por reactivos se mostrará cada una de las actividades descritas en el cuestionario, así como la frecuencia con la que, a juicio de los alumnos, el profesor las realizó.

Para interpretar los datos, el informe deberá mostrar lo siguiente:

Casi nunca	Algunas veces	Frecuentemente	Casi siempre	Siempre
1	2	3	4	5
0-19%	20-39%	40-59%	60-79%	80-100%

3.4 Planteamiento de la solución y posibles módulos

Se propone un nuevo sistema, el cual trabaje desde la web, que sea sencillo, seguro y eficaz para procesar la información derivada del proceso de evaluación.

De esta manera los alumnos serán responsables de ingresar al sistema para evaluar a sus profesores en el tiempo indicado desde cualquier computadora con acceso a internet y de forma objetiva y parcial.

El sitio principal del sistema de evaluación docente es el del alumno, desde el cual, el alumno que ingrese podrá evaluar a los profesores disponibles según el semestre y la asignatura.

El alumno sólo podrá evaluar a sus profesores, ingresando su número de cuenta y una contraseña. Después de mostrar las instrucciones el sistema mostrará una lista de las asignaturas y profesores que podrá evaluar. Eligiendo la materia a evaluar, el sistema cargará el cuestionario. Al finalizar la elección de respuestas podrá enviar su evaluación quedando almacenada en el sistema y a manera de confirmación, el sistema mostrará en pantalla la asignatura que ha sido evaluada.

Se puede representar el proceso de evaluación en línea mediante diagramas de bloques. En él se identifican tres procesos principales que puede realizar el alumno dentro del ambiente del sistema: validarse para poder acceder al sistema, evaluar a sus profesores una vez que se ha autenticado y la confirmación de que ha hecho la evaluación para salir del sistema.

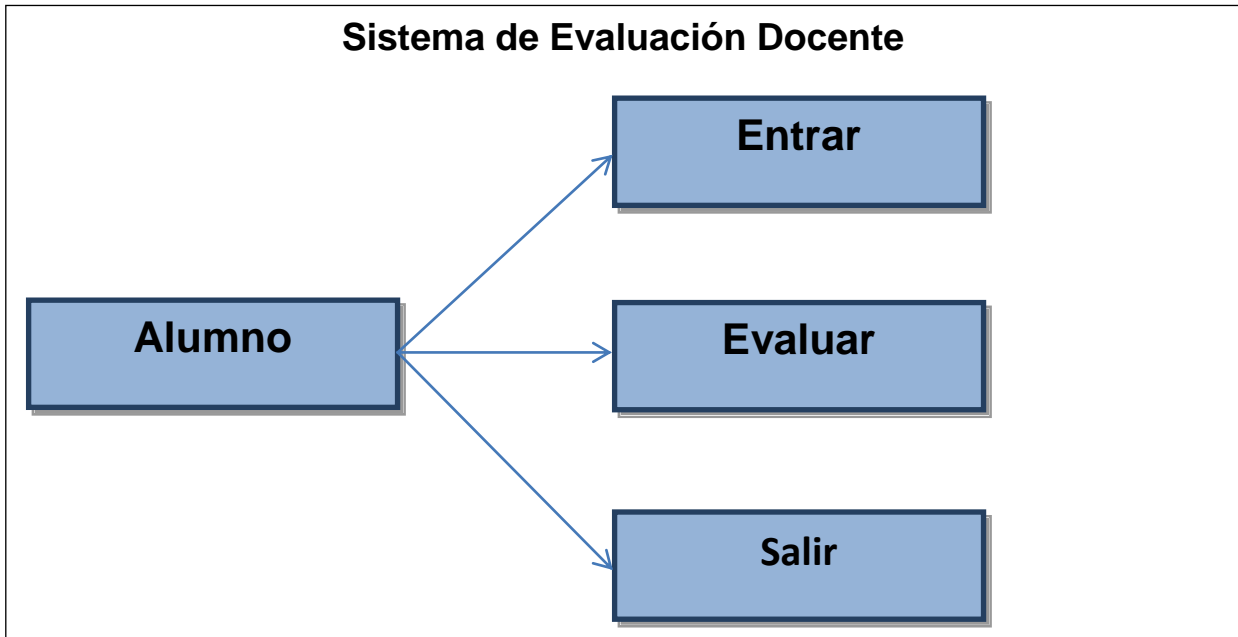


Figura 3.4.1. Diagrama de bloques para el usuario alumno.

En la primera pantalla, el sistema requiere que el alumno ingrese su número de cuenta y su contraseña, que en este caso es la misma que utiliza en el sistema de inscripciones, como se muestra en la figura 3.4.2.



Figura 3.4.2. Módulo de ingreso al sistema.

Si los datos del usuario son encontrados en la base de datos, el sistema mostrara la pantalla con instrucciones para poder iniciar con la evaluación (véase figura 3.4.3).

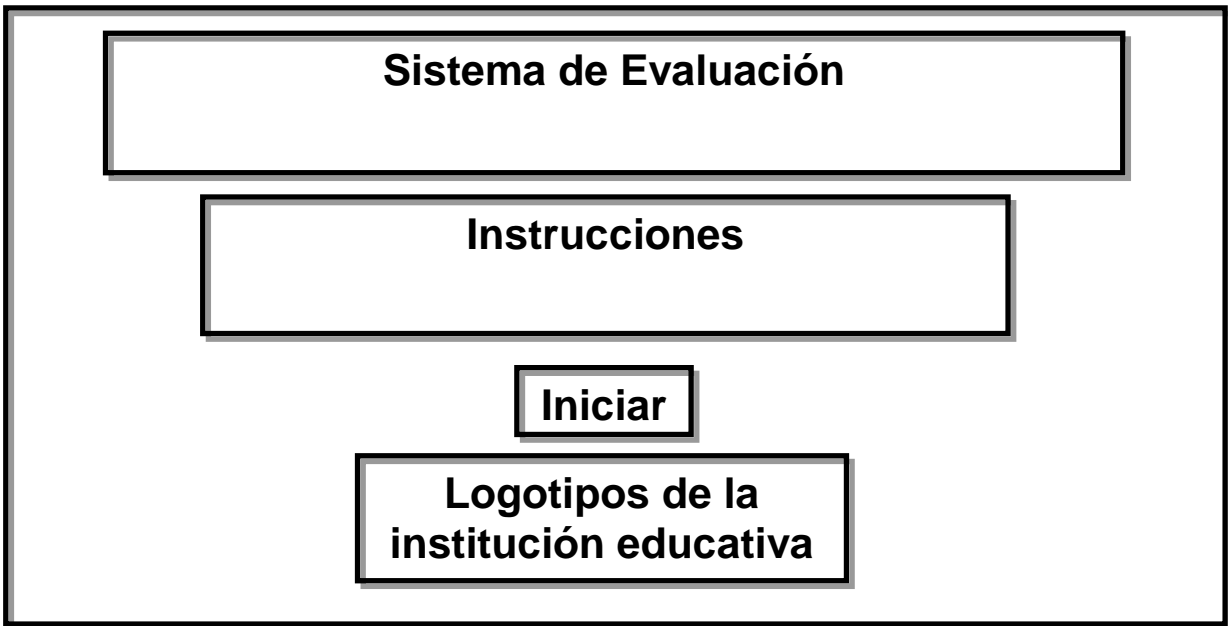


Figura 3.4.3 Instrucciones de la evaluación.

En la siguiente pantalla el alumno podrá observar las materias a calificar para ese periodo correspondiente (figura 3.4.4).

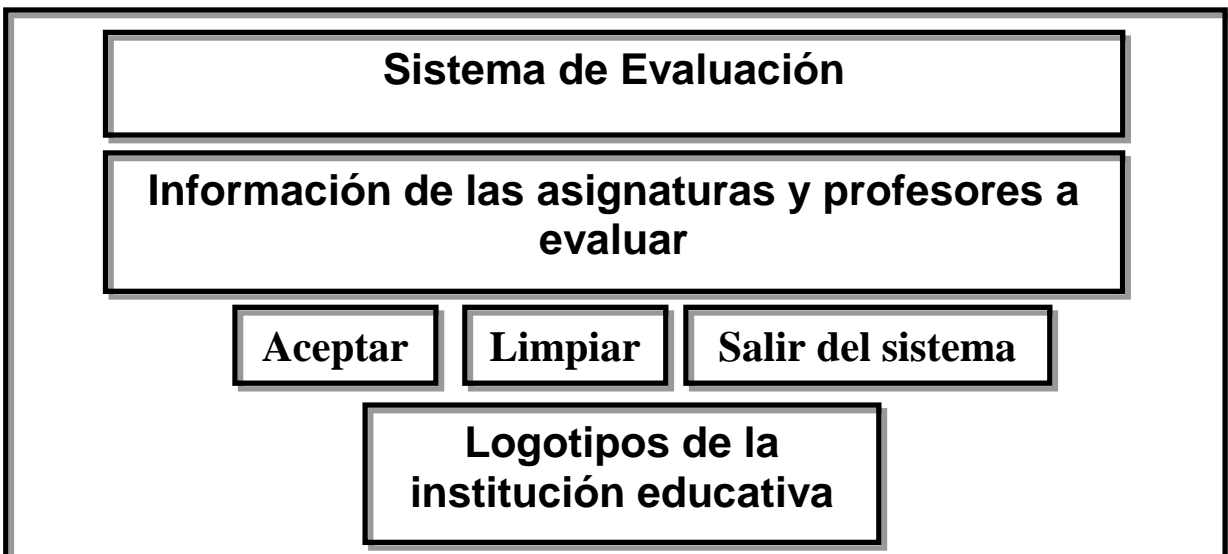


Figura 3.4.4 Elección de asignatura a evaluar.

A continuación podrá observarse el cuestionario con las preguntas de opción múltiple, al pulsar el botón siguiente aparecerá la siguiente parte del cuestionario y así sucesivamente hasta terminar con las preguntas (véase figura 3.4.5).

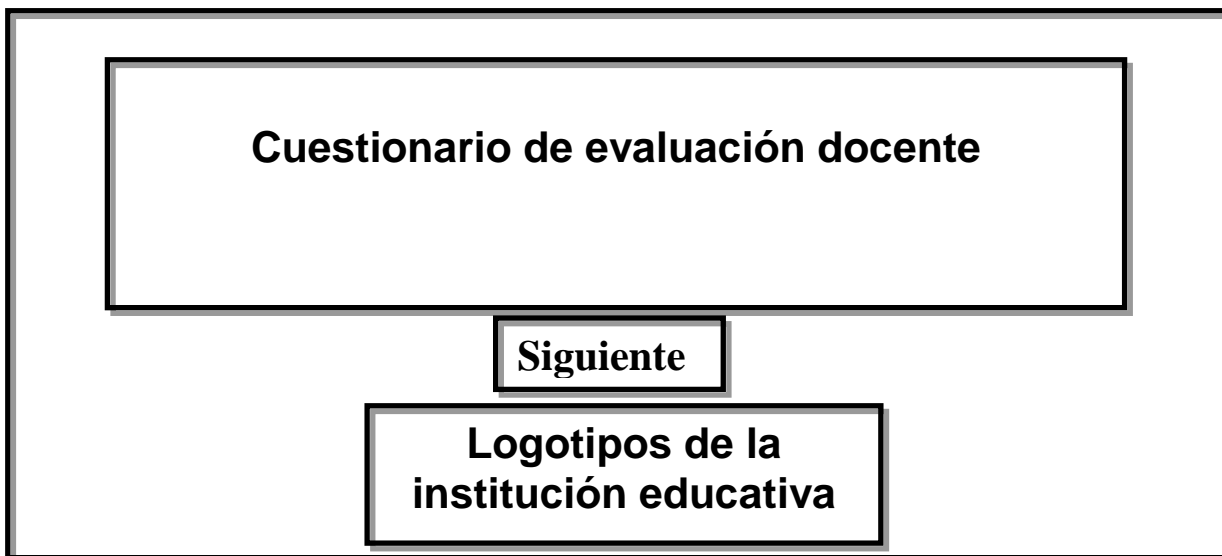


Figura 3.4.5. Cuestionario

En la última parte del cuestionario se deja un espacio para texto libre en donde el alumno puede ingresar sus comentarios, también se muestra el botón para dar por finalizada la evaluación (véase figura 3.4.6).

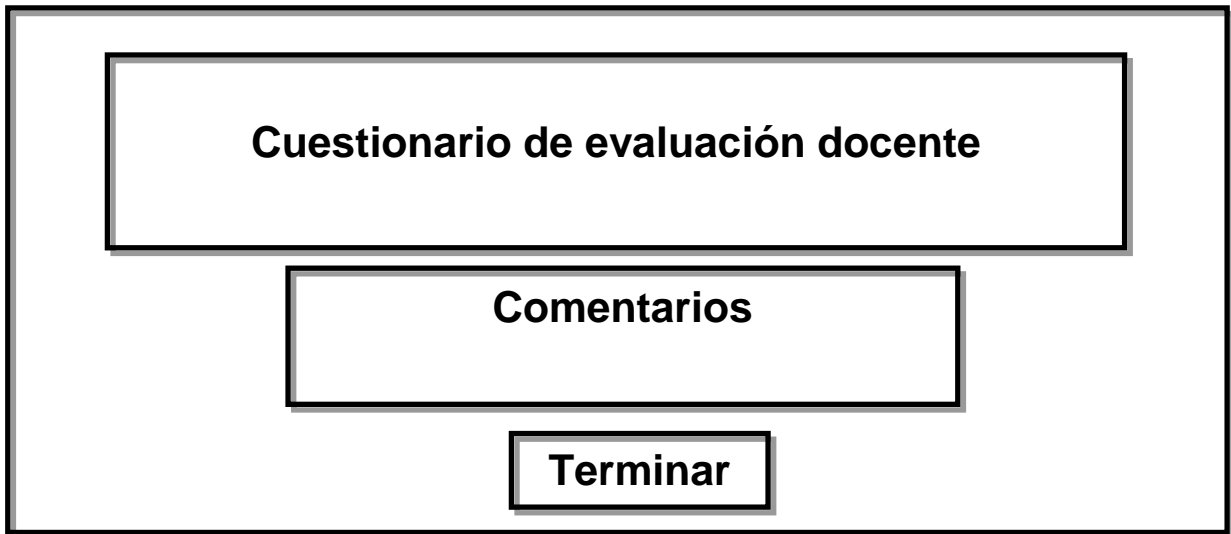


Figura 3.4.6. Final de la evaluación.

Una vez se ha terminado de evaluar a un profesor, el sistema confirmara si la evaluación se ha realizado con éxito y permitirá volver a la pantalla de elección de materia a evaluar para continuar con las demás materias o elegir salir del sistema (véase figura 3.4.4). En caso de haber terminado todas las asignaturas, el sistema mostrará un número de folio e indicará que se ha concluido la evaluación de todos los profesores.

3.5 Justificación de la metodología a utilizar

Las metodologías y estándares utilizados en un desarrollo de software nos proporcionan las guías para poder conocer todo el camino a recorrer desde antes de empezar la implementación, con lo cual se asegura la calidad del producto final, así como también el cumplimiento en la entrega del mismo en un tiempo estipulado.

Para la construcción de un sistema informático es de suma importancia elegir la metodología y las herramientas adecuadas. Actualmente donde el cambio y desarrollo tecnológico tanto en hardware como software, es constante, la oferta es muy amplia.

Para llevar a cabo una buena elección se tienen que tomar en cuenta diversos factores tales como el tamaño y tipo de la aplicación a desarrollar, el costo-beneficio, el número de usuarios que lo emplearan, el equipo disponible, el tiempo de entrega del sistema y otros factores adicionales. Tomando en cuenta estos factores se analizan algunas opciones de solución, para posteriormente seleccionar la que mejor se ajuste a los requerimientos del sistema que se desea desarrollar.

Elegir la metodología adecuada para el desarrollo de software determina el éxito o no del proyecto a realizar, es por ello que a continuación se detalla la que se considera más apropiada: RUP con UML, mismas que se han utilizado durante el desarrollo del Sistema, Además de RUP con UML existen otras metodologías que son importantes, aquí presentamos algunas.

- Desarrollo evolutivo. Este modelo es el desarrollo de una implantación del sistema inicial, exponerla a los comentarios del usuario, refinarla en N versiones hasta que se desarrolle el sistema adecuado.

- Modelo en cascada. Toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y las representa como fases separadas del proceso.
- Desarrollo formal de sistemas. Este modelo se basa en transformaciones formales de los requisitos hasta llegar a un programa ejecutable.
- Codificar y corregir. Consiste en implementar algo de código y luego pensar acerca de requisitos, diseño, validación, y mantenimiento.
- Desarrollo en espiral. El ciclo de desarrollo se representa como una espiral, en lugar de una serie de actividades sucesivas con retrospectiva de una actividad a otra.

Elección del Back-End (Base de Datos).

Para la elección de este componente se analizaron tres bases de datos: Todas son de distribución y utilización libre (gratuita).

PostgreSQL.

Sistema de base de datos objeto-relacional. El código fuente es libre.

Características principales.

- Corre nativamente sobre diversos sistemas operativos (Windows NT, XP, Linux, Solaris, etc.)
- Interfaces nativas para ODBC, JDBC, C, C++, PHP, Perl, TCL, Python y otros
- Solo permite conexiones desde el equipo local vía sockets de dominio.

- Permite cargar bloques de código para que se ejecuten en el servidor (Funciones). Estas son escritos en diferentes lenguajes, tales como Java, C, C++ y PL/pgSQL.
- Permite la alta concurrencia vía un sistema llamado MVCC (Acceso Concurrente Multiversión).
- Amplia variedad de tipos de datos permitiendo de manera adicional que los usuarios puedan crear sus propios tipos de datos.
- Uso de llaves foráneas, triggers, vistas, integridad transaccional entre otras.

MySQL

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Se ofrece con licenciamiento dual.

Características principales

- Provee diferentes APIs que permiten a aplicaciones escritas en diferentes lenguajes de programación, acceder a las bases de datos MySQL, incluyendo a C, C++, C#, Pascal; Delphi, Java, Lisp, Perl, PHP, Python y otros. También existe una interfaz ODBC, llamada MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos de MySQL.
- Es muy utilizado en aplicaciones Web como MediaWiki o Drupal en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python). Es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación.
- MySQL funciona sobre múltiples plataformas, incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64,

- Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows.
- Un amplio subconjunto de ANSI SQL 99, y varias extensiones.
- Soporte a multiplataforma.
- Procedimientos almacenados, Triggers y Cursores.
- Vistas actualizables.
- Soporte X/Open XA de transacciones distribuidas.
- Motores de almacenamiento independientes.
- Transacciones con los motores de almacenamiento InnoDB, BDB Y Cluster.
- Puntos de recuperación (savepoints) con InnoDB.
- Sub-SELECTs (o SELECTs anidados).
- Indexamientos y búsqueda de campos de texto completos usando el motor de almacenamiento MyISAM.
- Embedded database library.
- Soporte completo para Unicode.
- Conforme a las reglas ACID usando los motores InnoDB, BDB y Cluster.
- En Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos.

Firebird

Es un software de administración de una base de datos relacional, similar en propósito a productos tales como DB2; Oracle ó PostgreSQL. Tiene dos componentes principales el servidor de base de Datos, que vive en la misma máquina que la Base de Datos, y la interface de aplicación, comúnmente referida como la Librería del Cliente.

Características principales

- Lenguaje y Soporte completo para Procedimientos Almacenados y Triggers
- Las Transacciones son totalmente ACID compliant
- Integridad referencial
- Arquitectura Multi Generacional
- Muy bajo consumo de recursos
- Soporte para Funciones externas (UDFs)
- Opción a usar la versión embebida - de un solo fichero - ideal para crear CDROM con catálogos, versiones de evaluación o monousuario de aplicaciones
- Escritura segura - recuperación rápida sin necesidad de logs de transacciones
- Muchas formas de acceder a las bases de datos: nativo/API, driver dbExpress, ODBC, OLEDB, .Net provider, driver JDBC nativo de tipo 4, módulo para Python, PHP, Perl, etc.
- Soporte nativo para los principales sistemas operativos, incluyendo Windows, Linux, Solaris, MacOS y otros.
- Backups incrementales
- Disponible para arquitecturas de 64bits
- Altas volúmenes de almacenamiento
- Completa implementación de cursores en PSQL

La tabla 3.5.1 muestra la comparación de algunas características de las tres Bases de datos anteriormente mencionadas.

CARACTERISTICAS	FIREBIRD 2.0	MYSQL 5.0	POSTGRESQL 8.1
Licenciamiento	Totalmente libre	Modelo dual	Totalmente libre
Plataformas	Larga lista de plataformas	Larga lista de plataformas	Larga lista de plataformas

Soporte de 64 bits	Si	Si	Si
Limite por Base de Datos	Limitado	Ilimitado	Ilimitado
Limite por Tabla	Limitados	16 TB en Solaris	32 TB
Número de índices por tabla	Limitados	64	Ilimitados
Limites numéricos	Enteros 64 bits	Enteros 64 bits 38 dígitos para numéricos y decimales	Precisión de 1000 dígitos
Limites de Cadenas (VARCHAR)	32765 bytes	65,532 bytes	1 Gb
Soporte de Unicode	Si (UTF-8 y 16)	Si (UTF-8)	Si (UTF-8)
Transacciones	Si, conformidad ACID	Si, conformidad ACID	Si, conformidad ACID
Candados a nivel renglón de Tabla (conurrencia)	Modelo MVCC	Modelo MVCC solo para tablas InnoDB	Modelo MVCC
Funcionalidad de TIMEOUT	No soportada	No soportada	Si soportada
Soporte de stores procedures	Soportada	Soportada, con restricciones	Soportada
Soporte de Triggers	Soportada	Soportada, con restricciones	Soportada
Escalabilidad/Soporte SMP	Soporte solo en el clasic Server	Soportada	Soportada
Balanceo de cargas	No soportado	Soporte parcial	Soportado
Replicación	Soportado	Soportado	Soportado

Respaldo en Línea (Online Backup)	Soportada con comando en línea	Soportado con la utilidad Mysqldump	Puede realizar backups continuos
--------------------------------------	--------------------------------	-------------------------------------	----------------------------------

Tabla 3.5.1 Software de Base de Datos.

Elección del FRONT-END

Para el desarrollo y construcción del front-end que es la parte que sirve de interfaz al usuario y operador, esta interfaz tiene que ser amigable y en un ambiente totalmente gráfico para la representación de resultados, debe ser operablemente sencilla capaz de realizar las acciones con la base de datos de una manera eficiente, para ello se han tomado los siguientes productos, PHP, JavaScript. Ver Tabla 3.5.2

PHP

Es un lenguaje de programación de uso general de script del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. PHP puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo.

Características de PHP

- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas, ejemplo que se hace evidente en el uso de php arrays.
- El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos. Incluso aplicaciones como Zend framework, empresa que desarrolla PHP, están totalmente desarrolladas mediante esta metodología.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).
- Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable.
- La principal ventaja se basa en ser un lenguaje multiplataforma.

- Capacidad de conexión con la mayoría de los manejadores de base de datos que
- se utilizan en la actualidad.
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos
- (llamados ext's o extensiones).
- Posee una muy buena documentación en su página oficial.
- Es Libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos. (Clases y herencia.).
- Muy sencillo de aprender.
- Similar en sintaxis a C y a PERL
- Manejo de excepciones
- Biblioteca nativa de funciones
- Destacada conectividad con MySQL.

JavaScript

JavaScript es un lenguaje de programación orientado a objetos, basado en prototipos, imperativo y dinámico.

Características de JavaScript:

- Orientado a objetos, se refiere a un método de programación y al diseño del lenguaje. Diseñar el software de forma que los distintos tipos de datos que usen estén unidos a sus operaciones.

- Creación de entidades más genéricas (objetos) que permitan la reutilización del software entre proyectos, una de las premisas fundamentales de la Ingeniería del Software.
- La independencia de la plataforma, significa que programas escritos en el lenguaje Java pueden ejecutarse igualmente en cualquier tipo de hardware. Este es el significado de ser capaz de escribir un programa una vez y que pueda ejecutarse en cualquier dispositivo.
- La licencia sobre Java de Sun insiste que todas las implementaciones sean “compatibles
- Las primeras implementaciones del lenguaje usaban una máquina virtual interpretada para conseguir la portabilidad.
- La portabilidad, es de hecho posible escribir programas para la plataforma Java que actúen de forma correcta en múltiples plataformas de distinta arquitectura,
- El concepto de independencia de la plataforma de Java cuenta, sin embargo, con un gran éxito en las aplicaciones en el entorno del servidor, como los Servicios Web, los Servlets, etc.
- En Java el problema fugas de memoria se evita en gran medida gracias a la recolección de basura (o automatic garbage collector). El programador determina cuándo se crean los objetos y el entorno en tiempo de ejecución de Java (Java runtime) es el responsable de gestionar el ciclo de vida de los objetos, el recolector de basura de Java permite una fácil creación y eliminación de objetos y mayor seguridad.
- La sintaxis de Java se deriva en gran medida de C++. Pero a diferencia de éste, que combina la sintaxis para programación genérica, estructurada y orientada a objetos, Java fue construido desde el principio para ser completamente orientado a objetos.
- Es seguro y fiable. Por lo que puede ser filtrado para el mismo Javascript. La seguridad es casi total.
- Los script tienen capacidades limitadas, por razones de seguridad, por lo cual no es posible hacer todo con Javascript, sino que es necesario usarlo

conjuntamente con otros lenguajes evolucionados, posiblemente más seguros, como Java.

- El código Javascript se ejecuta en el cliente por lo que el servidor no es solicitado más de lo debido.
- Reducción de código
- Código más conciso y de fácil lectura
- Facilidad en manejo de arrays
- Reutilización de código y la capacidad de abstraerlo en capas.

Servidor Apache

Apache Tomcat es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual.

Características del Servidor Apache:

- Alta configurabilidad y robustez, siendo además casi universal al estar en una multitud de Sistemas Operativos.
- Tecnología gratuita de código fuente abierto le da una transparencia inmediata, de forma que si quisiéramos ver qué es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto.
- Trabaja con Perl, PHP y otros lenguajes de script, Java y páginas JSP, teniendo todo el soporte que se necesita para tener páginas dinámicas.
- Por su diseño modular es muy sencillo ampliar las capacidades del servidor web Apache.
- Se puede personalizar la respuesta ante los posibles errores que se den en el servidor, por supuesto se puede configurar para que ejecute un determinado script cuando ocurra un error marcado.

- Alta configurabilidad en la creación y gestión de logs, nos permite la creación de ficheros de log a medida del administrador, teniendo así el control máximo sobre lo que pasa en el servidor.
- Características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración. Ver Tabla 3.5.2

CARACTERISTICAS	PHP	JavaScript
Marca	Libre	Sun
Programación	Orientada a objetos	Orientada a objetos
Conectividad con Base de Datos	Todas	Todas
Herramienta de generación de SQL	Sí	Sí
Plataforma de desarrollo recomendada	Windows XP Hacia Arriba	Windows XP Hacia Arriba
Gestión de excepciones	Sí	Sí
Ventajas	Multiplataforma, software libre	Independencia de plataforma y Portabilidad

Tabla 3.5.2 Software Front-End

Podemos concluir con respecto al entorno de desarrollo y al manejador de base de datos, las herramientas que utilizaremos serán PHP, JavaScript, PostgreSQL, y como servidor Apache Tomcat, porque son software de uso libre y no se requieren licencias para su empleo, mismos que son compatibles entre sí y la metodología a utilizar será RUP ya que es la que mejor se adapta a las

necesidades del sistema en desarrollo y con UML se facilitará el proceso de modelado para un mejor entendimiento durante el desarrollo del sistema.

La elección de las herramientas para el desarrollo del sistema se realizó tomando en cuenta las siguientes consideraciones:

- Los conocimientos con los que cuentan los integrantes del equipo desarrollador.
- La portabilidad del software.
- El uso de Software de distribución Gratuita y Libre.

Las herramientas elegidas para el desarrollo del sistema son las siguientes que veremos en la tabla 3.5.3:

Componente	Producto	Característica Principal
Manejador de Base de Datos	PostgreSQL	Alta concurrencia, Instalación Ilimitada, Soporte, Ahorros considerables en costos de operación, Estabilidad y Confiabilidad Legendarias, Extensible, Multiplataforma, Diseñado para ambientes de alto volumen, Herramientas gráficas de diseño y administración de BD
Servidor de Aplicaciones Web	Apache / Tomcat	Soporte de tecnología Java (Servlet y JavaServer Pages), Altamente configurable, Estabilidad, Independencia de la plataforma, Código abierto, Administración
Lenguaje de Programación para páginas dinámicas, acceso y actualización de Base de datos, Interfaz	PHP, Java Server Pages (JSP)	Código escrito en Java y HTML, Independencia de Plataforma, portabilidad, Facilidad de instalación configuración y utilización

Tabla 3.5.3 Software Utilizado

CAPITULO IV. DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN

4.1 Modelado del Sistema

Para desarrollar el sistema de una forma profesional es indispensable seguir una metodología con el fin de obtener un software de calidad y que sea una buena solución al problema. El modelado es importante porque con él podemos explicar el comportamiento del sistema, además de tener un mayor control sobre el mismo. Podemos tener modelos que proporcionan información desde la más general hasta las más detalladas.

Para nuestro sistema, se ha elegido trabajar con los siguientes diagramas que proporciona el Lenguaje de Modelado Unificado (UML) y basado en la Metodología RUP:

- Diagrama de Casos de Uso
- Diagrama de Modelado de Datos
- Diagrama de Entidad – Relación
- Diagrama de Clases
- Diccionario de Datos

4.1.1 Diagrama de Casos de Uso

Un Diagrama de Casos de Uso muestra la relación entre los actores y los casos de uso del sistema. Básicamente es una representación de la funcionalidad del sistema con respecto a los usuarios. Vea figura 4.1.1

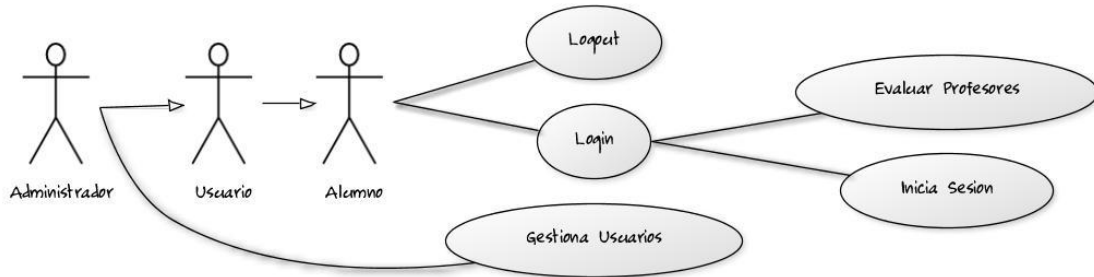


Fig. 4.1.1 Los actores serán el administrador y los usuarios. El administrador gestionará tanto el sistema, los usuarios y la información. Los alumnos harán uso del sistema

Casos de Uso – Usuario Registrado. Vea figura 4.1.2.

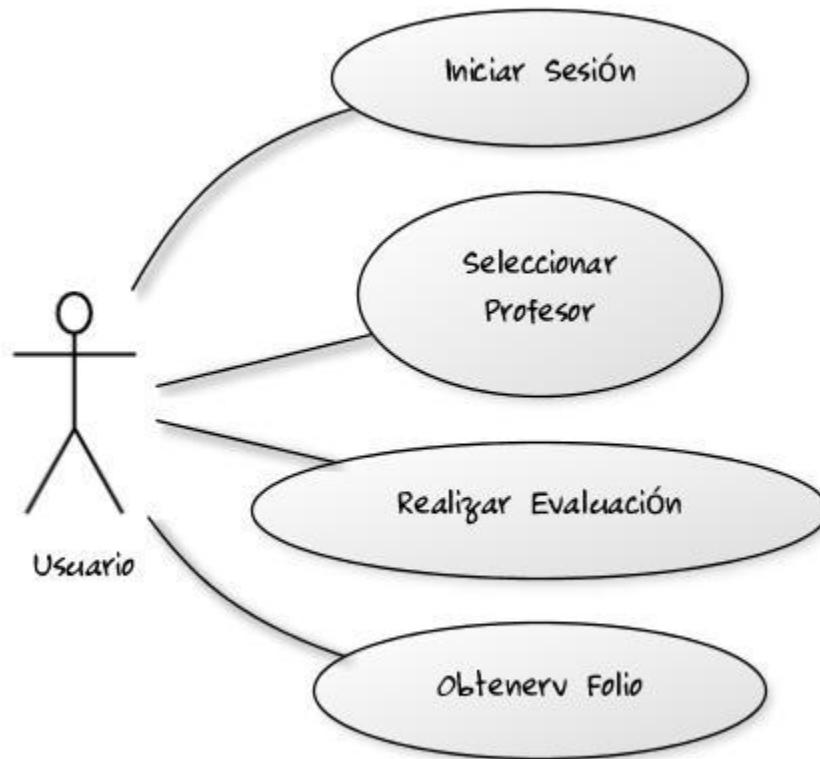


Fig. 4.1.2 El usuario que esté registrado podrá iniciar sesión en el sistema, evaluar a sus profesores del semestre actual, así como obtener un folio que demuestre que la realizó.

Caso de Uso	Usuario			
Actores	Alumnos			
Actor		Sistema		
Paso	Acción	Paso	Acción	Excepción
1	Iniciar Sesión	2	Comprobar en la Base de Datos y traer información del alumno	
3	Seleccionar profesor a evaluar	4	Registrar Evaluación del Profesor y Asignatura seleccionadas	
5	Evaluar todos los profesores	6	Generarle un número de folio	

Casos de Uso – Administrador. Vea figura 4.1.3.

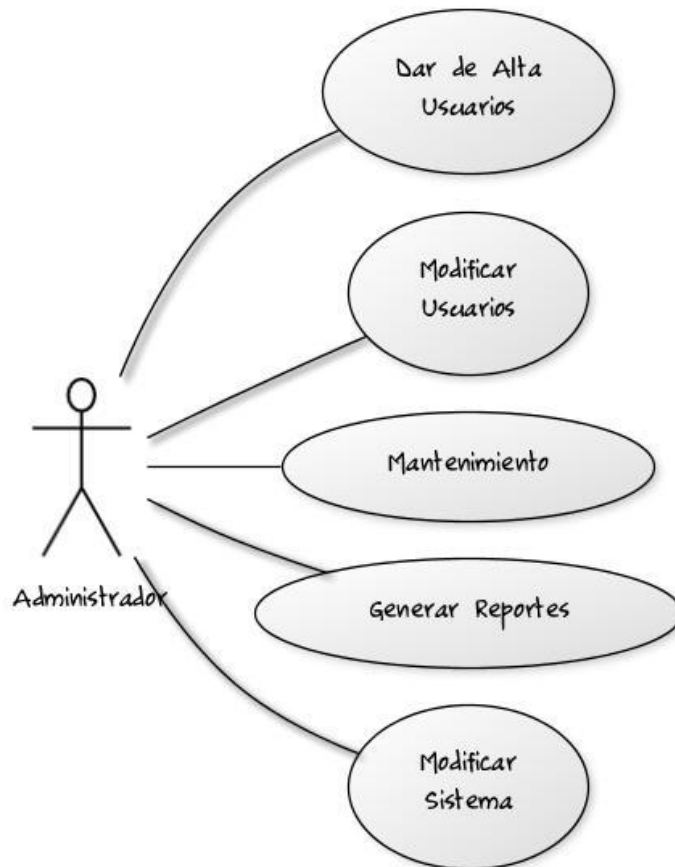


Figura 4.1.3

El administrador mantendrá en buen funcionamiento el sistema, procesará la información para la generación de reportes, gestionará a los usuarios y hará las modificaciones necesarias al sistema de ser requeridas.

Caso de Uso	Generación Reportes				
Actores	Administrador				
Actor			Sistema		
Paso	Acción	Paso	Acción	Excepción	
1	Ejecutar el programa				
2	Seleccionar tipo de reporte	3	Generar Reporte		

4.1.2 Diagrama de Modelado de Datos

Un modelo de datos es un conjunto de herramientas conceptuales para describir datos, sus relaciones, su significado y sus restricciones de consistencia.

Hay 3 tipos de modelado de datos:

- Conceptual: Modelo muy general y abstracto. Es una visión muy general del negocio. Describen las estructuras de datos y restricciones de integridad. Se utilizan durante la etapa de análisis de un problema y están orientados a representar los elementos que intervienen y sus relaciones.
- Lógico: Es una versión más completa que incluye todos los detalles acerca de los datos.
- Físico: Es un esquema que se implementará en un manejador de base de datos.

4.1.3 Diagrama Entidad-Relación

El Modelo de Entidad Relación es un modelo de datos que consiste en un conjunto de objetos básicos llamados entidades y relaciones.

Una entidad es cualquier objeto real, que existe en cierto contexto o puede llegar a existir y del cual deseamos guardar información o sus atributos, que son las características o propiedades asociadas a la entidad que toman valor en una instancia particular.

Básicamente se compone de lo siguiente:

Entidades: Todo lo que existe y puede ser capaz de ser descrito. Puede ser física (persona, producto o algo que sea tangible) o puede ser un elemento de tipo conceptual (empresa, curso o algo que sea intangible).

Atributos: Es una característica de una entidad que puede identificar, relacionar o describir.

Relaciones: Es la conexión entre 2 entidades.

Cardinalidad: Es el número de ocurrencias que puede existir entre un par de entidades

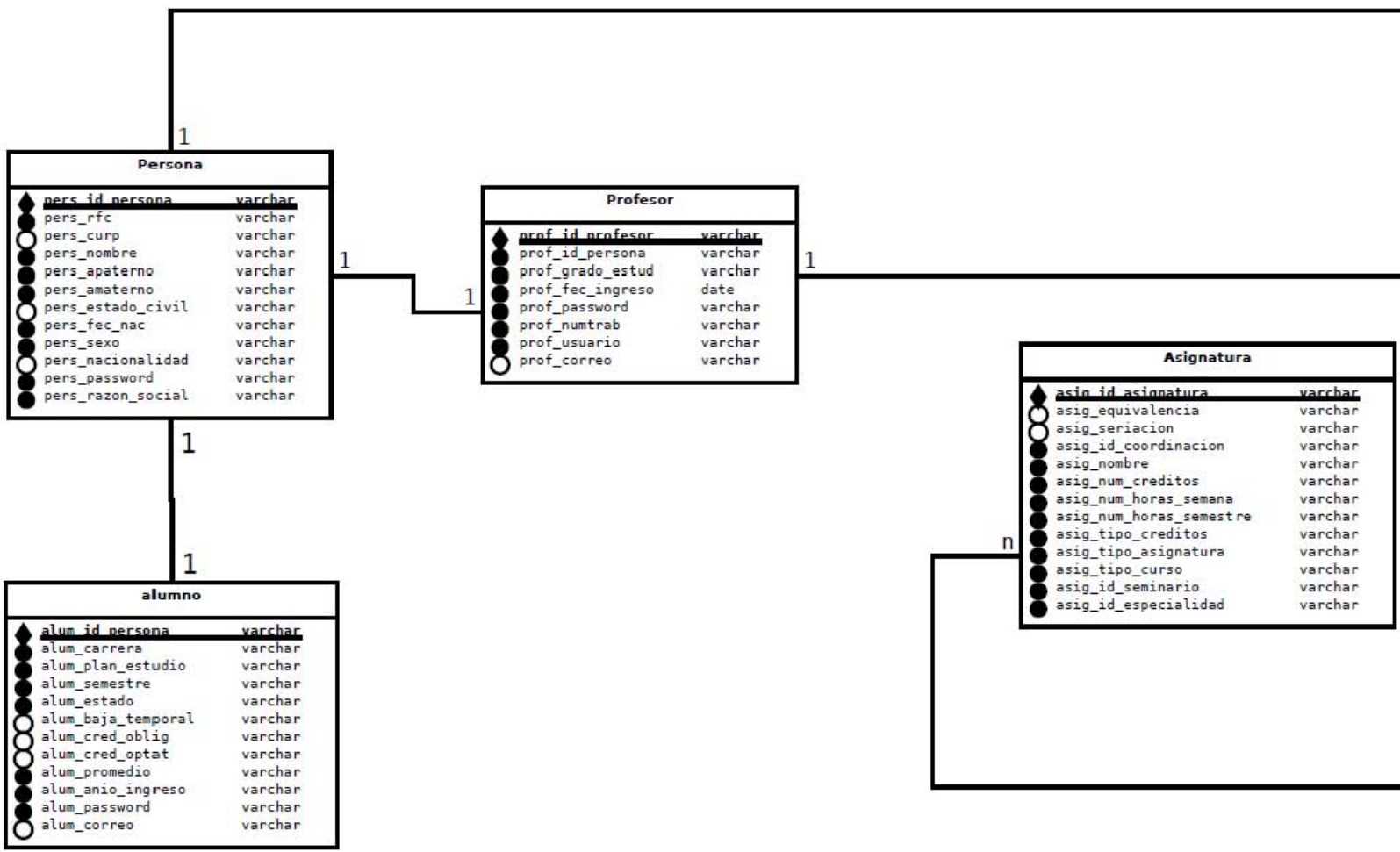
Llave primaria: Sirve para identificar a los elementos de un conjunto de entidades.

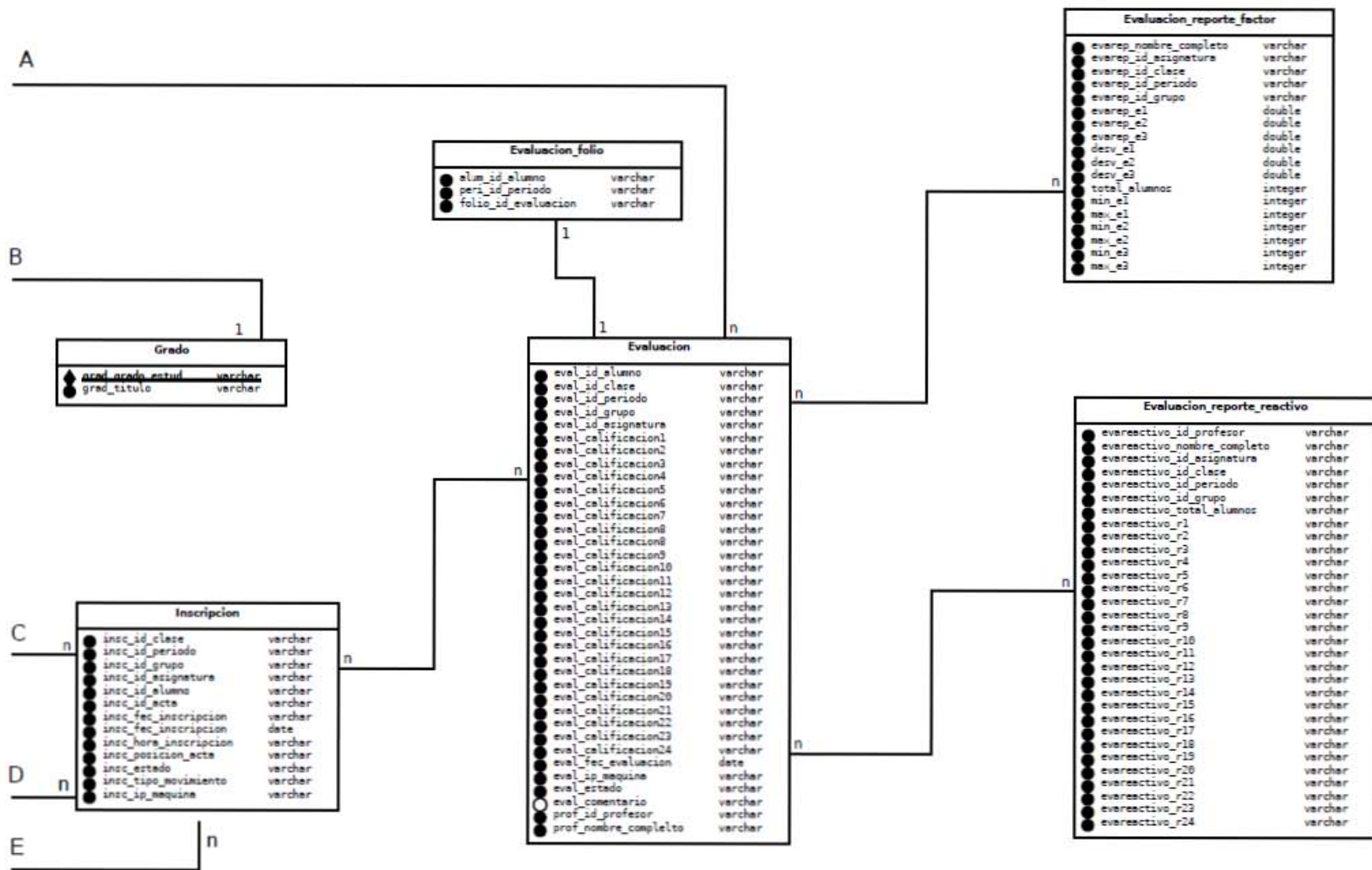
Vea figura 4.1.2.1.

A

B

C





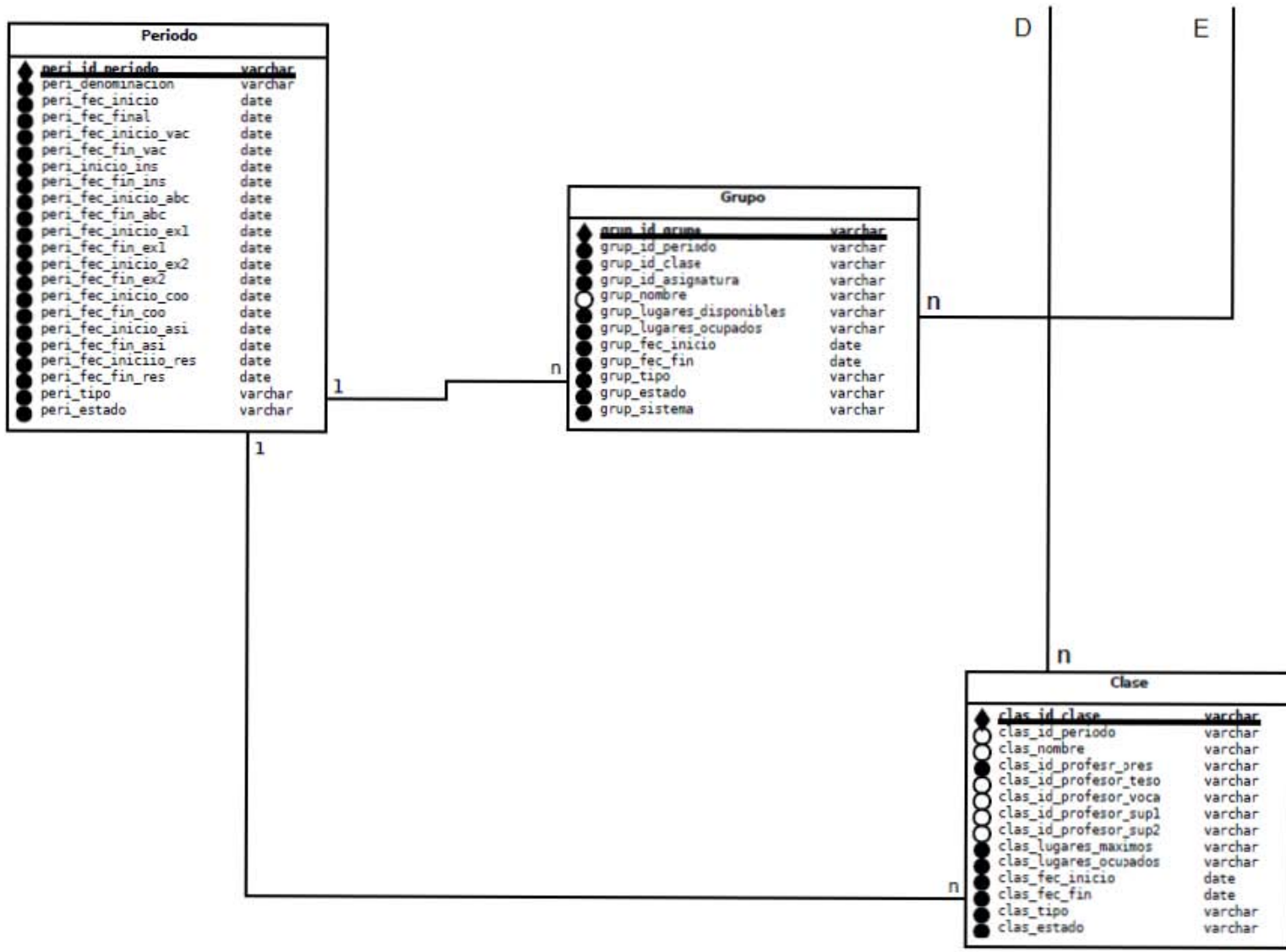


Figura 4.1.2.1. Diagrama Entidad-Relación

4.1.4 Diagrama de clases

Representa los objetos fundamentales del sistema, es decir, los que percibe el usuario y con los que espera tratar para completar su tarea en vez de objetos del sistema o de un modelo de programación.

En UML, una clase es representada por un rectángulo que posee 3 divisiones. Vea figura 4.1.4.1:

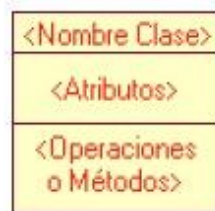


Figura 4.1.4.1. Ejemplo Diagrama de Clases

En donde:

- Nivel Superior: Contiene el nombre de la Clase
- Nivel Intermedio: Contiene los atributos que caracterizan a la Clase.
- Nivel Inferior: Contiene los métodos u operaciones, los cuales son la forma cómo interactúan el objeto con su entorno.

La clase define el ámbito de definición de un conjunto de objetos. Cada objeto pertenece a una clase. Los objetos se crean por instanciación de las clases.

Los atributos de la clase son los siguientes:

- Tipo: Puede llegar a depender del lenguaje de programación a utilizar
- Valor inicial: valor que poseerá el atributo al crear el objeto.
- Visibilidad: está relacionado con el encapsulamiento.

- Multiplicidad: determinar si un atributo debe estar o no, y si posee un único valor o una lista de valores.
- Ordenamiento: especifica si el atributo determina alguna relación de orden dentro de la clase.
- Capacidad de cambio: permite definir atributos con valores constantes.
- Modificadores: un atributo puede ser de clase, derivado, volátil, transitorio.
- Niveles de encapsulamiento:
- (-) Privado: Solo es visible dentro de la clase.
- (#) Protegidos: Los atributos/operaciones protegidos solo son visibles por clases derivadas de la original
- (+) Públicos: Son visibles para todas las clases.

A continuación presentamos los diagramas de clase para las entidades que representan el llenado de la encuesta por parte del alumno. Vea figura 4.1.4.2:

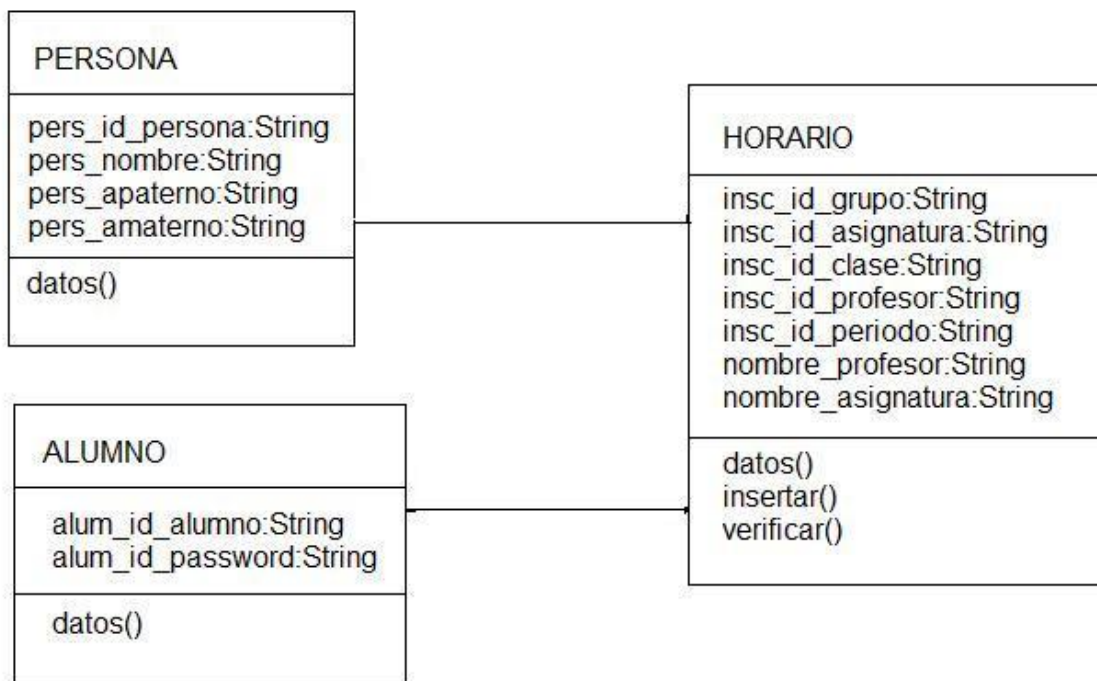


Figura 4.1.4.2 Diagrama de clases del Sistema

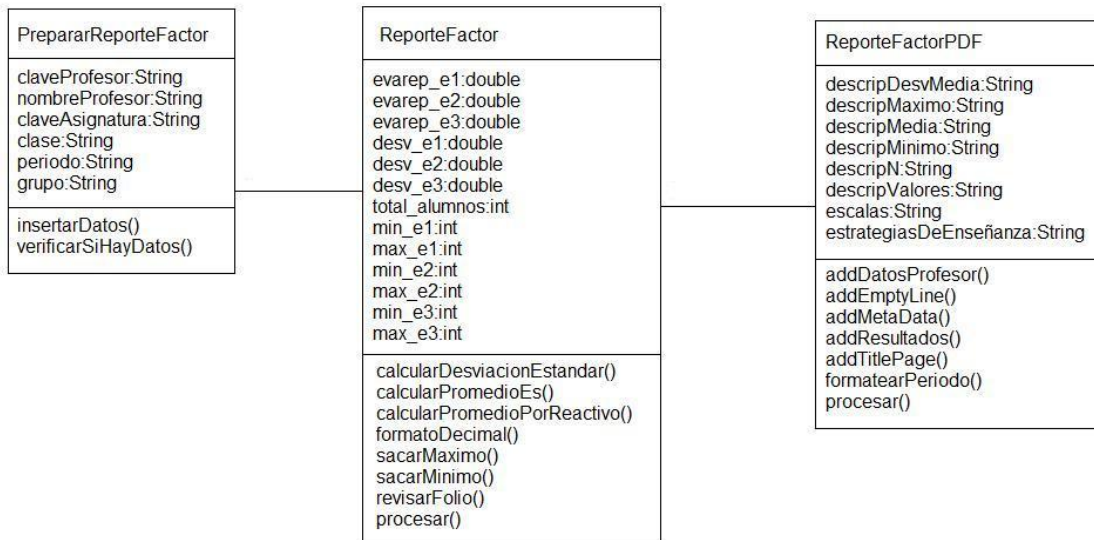


Figura 4.1.4.3

Diagrama Generación de Reporte por Factor

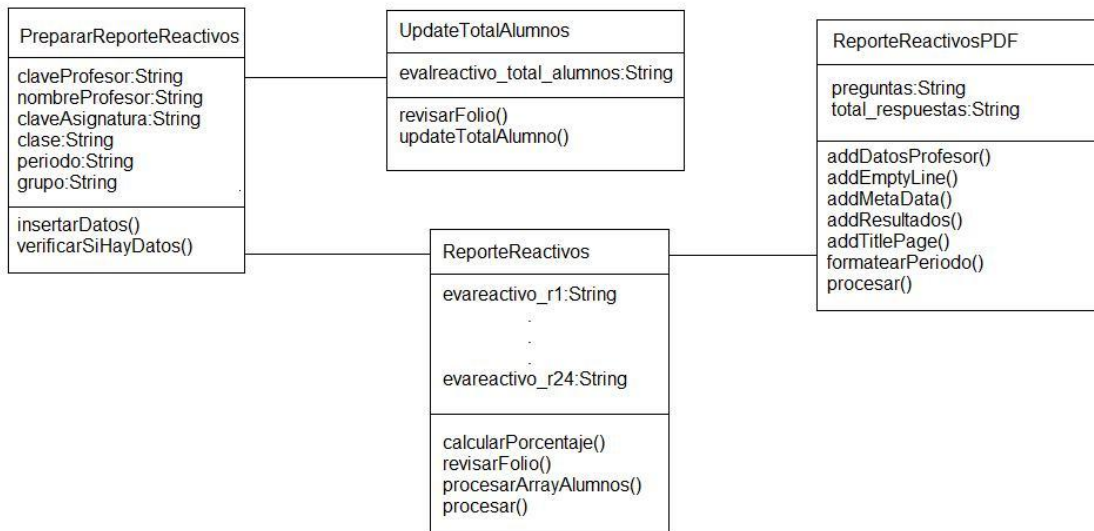


Figura 4.1.4.4

Diagrama Generación de Reporte por Reactivos

4.1.5 Diccionario de Datos

Contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema, incluyendo nombre, descripción, alias, contenido y organización. Aquí se documenta la información generada por el diseñador de la base de datos. Entre los tipos de información que el diccionario debe de almacenar están:

- Nombre de los atributos de cada relación
- Dominio de los atributos
- Restricciones de integridad de cada relación

Se utiliza para:

- Describir el propósito de la Base de Datos.
- Descripción detallada de cada tabla dentro de la Base de Datos.
- Documental la estructura de cada tabla, incluyendo sus campos y sus tipos de datos con comentarios, sus índices y vistas.
- Descripción del uso de valores no nulos y únicos.

Ver figuras 4.1.5.1, 4.1.5.2, 4.1.5.3, 4.1.5.4, 4.1.5.5, 4.1.5.6, 4.1.5.7, 4.1.5.8, 4.1.5.9, 4.1.5.10, 4.1.5.11, 4.1.5.12 y 4.1.5.13.

Persona						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	pers_id_persona	Not null	18	pk	Alumno, profesor, evaluacion	Id de la persona
rfc	pers_rfc	Not null	13			Rfc de la persona
curp	pers_curp	null	18			Curp de la

						persona
nombre	pers_nombre	Not null	30			Nombre
Apellido paterno	pers_apaterno	null	25			Apellido Paterno
Apellido materno	pers_amaterno	null	25			Apellido Materno
Estado civil	pers_estado_civil	Not null	1			Estado Civil
Fecha nacimiento	pers_fec_nac	Not null				Fecha Nacimiento
Sexo	perssexo	Not null	1			Sexo
Nacionalidad	pers_nacionalidad	Not null	1			Nacionalidad
Password	pers_password	Not null	50			Password
Razón Social	pers_razon_social	Not null	2			Razón Social

Figura 4.1.5.1. Diccionario de Datos Tabla Persona

Alumno						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	alum_id_alumno	Not null	13	Pk		Id alumno
Id persona	alum_id_persona	Not null	18	fk	Persona	Id persona
Carrera	alum_carrera	Not null	3			Carrera
Plan de Estudios	alum_plan_estudio	Not null	4			Plan de estudios
Semestre	alum_semestre	Not null	2			Semestre actual
Estado	alum_estado	Not null	1			Activo / inactivo
Baja Temporal	alum_baja_temporal	Null	1			Baja temporal
Créditos Obligatorios	alum_cred_oblig	null	3			Créditos obligatorios
Créditos Optativos	alum_cred_optat	Null	3			Créditos Optativos
Promedio	alum_promedio	Null	6			Promedio
Año de ingreso	alum_anio_ingreso	Not null	4			Año de ingreso
Password	alum_password	Null	50			Password

Correo	alum_correo	null	32			Correo institucional
--------	-------------	------	----	--	--	----------------------

Figura 4.1.5.2. Diccionario de Datos Tabla Persona

Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	prof_id_profesor	Not null		Pk	Evaluación	Id profesor
Id persona	prof_id_persona	Not null		fk	Evaluación, persona	Id persona
Grado de Estudios	prof_grado_estud	Not null				Grado de Estudios
Fecha de ingreso	prof_fec_ingreso					Fecha de ingreso
Password	prof_password					Password
Número de trabajador	prof_numtrab					Número de trabajador
Usuario	prof_usuario					Nombre de usuario
Correo	prof_correo					Correo institucional

Figura 4.1.5.3. Diccionario de Datos Tabla Profesor

Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	asig_id_asignatura	Not null	10	pk	Evaluación	Id asignatura
Equivalencia	asig_equivalencia		10			Equivalencia
Seriación	asig_seriacion		10			Seriación
Id Coordinación	asig_id_coordinacion	Not null	4			Id coordinación
Nombre Asignatura	asig_nombre	Not null	100			Nombre de la asignatura
Créditos	asig_num_creditos	Not null	2			Créditos de la asignatura

Horas a la Semana	asig_num_horas_semana	Not null	3			Horas a la semana
Hora al Semestre	asig_num_horas_semestre	Not null	3			Horas al semestre
Tipo Créditos	asig_tipo_creditos	Not null	1			Obligatorios/optativos
Tipo Asignatura	asig_tipo_asignatura	Not null	1			Obligatoria/optativa
Tipo Curso	asig_tipo_curso	Not null	1			Ordinario/internsemestral
Seminario	asig_id_seminario		3			Id seminario
Especialidad	asig_id_especialidad		3			Id especialidad

Figura 4.1.5.4. Diccionario de Datos Tabla Asignatura

Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	peri_id_periodo	Not null	6	pk	Evaluacion	Id
Denominacion	peri_denominacion	Not null	30			Denominacion
Inicio	peri_fec_inicio	Not null	date			Inicio
Fin	peri_fec_fin	Not null	date			Fin
Inicio vacaciones	peri_fec_inicio_vac		date			Inicio vacaciones
Fin vacaciones	peri_fec_fin_vac		date			Fin vacaciones
Inicio inscripciones	peri_fec_inicio_ins		date			Inicio inscripciones
Fin inscripciones	peri_fec_fin_ins		date			Fin inscripciones
Inicio altas, bajas, cambios	peri_fec_inicio_abc		date			Inicio altas, bajas, cambios
Fin altas, bajas, cambios	peri_fec_fin_abc		date			Fin altas, bajas, cambios
Inicio extraordinarios	peri_fec_inicio_ex1		date			Inicio extraordinarios
Fin extraordinarios	peri_fec_fin_ex1		date			Fin extraordinario

						s
Inicio extraordinarios	peri_fec_inicio_ex2		date			Inicio extraordinarios
Fin extraordinarios	peri_fec_fin_ex2		date			Fin extraordinarios
Inicio cambio coordinación	peri_fec_inicio_coo		date			Inicio cambio coordinación
Fin cambio coordinación	peri_fec_fin_coo		date			Fin cambio coordinación
Inicio asistencia	peri_fec_inicio_asi		date			Inicio asistencia
Fin asistencia	peri_fec_fin_asi		date			Fin asistencia
Inicio reservación	peri_fec_inicio_res		date			Inicio reservación
Fin reservación	peri_fec_fin_res		date			Fin reservación
Tipo	peri_tipo		date			Ordinario/Extraordinario
estado	peri_estado		date			Activo/Inactivo

Figura 4.1.5.5. Diccionario de Datos Tabla Periodo

Grupo						
Nombre	Acónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	grup_id_grupo	Not null	10	Pk	Evaluación	Id grupo
Id periodo	grup_id_periodo	Not null	6	fk	Periodo, evaluación	Id periodo
Id clase	grup_id_clase	Not null	10	Fk	Clase, evaluación	Id clase
Id asignatura	grup_id_asignatura	Not null	10	fk	Asignatura, evaluación	Id asignatura
nombre	grup_nombre		75			Nombre
Lugares Disponibles	grup_lugares_disponibles	Not null	3			Lugares Disponibles
Lugares	grup_lugares_ocupados	Not null	3			Lugares

Ocupados						Ocupados
Fecha de inicio	grup_fec_inicio	Null	date			Fecha de inicio
Fecha de fin	grup_fec_fin	Null	date			Fecha de fin
Tipo	grup_tipo	Not null	2			Activo / inactivo
Estado	grup_estado	Not null	2			Estado
Sistema	grup_sistema	null	1			Sistema

Figura 4.1.5.6. Diccionario de Datos Tabla Grupo

Clase						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id	clas_id_clase	Not null	10	Pk	Evaluación	Id clase
Id periodo	clas_id_periodo	Not null	6	fk	Periodo	Id periodo
Nombre	clas_nombre	null	70			Nombre
Profesor titular	clas_id_profesor_pres	Not null	13			Profesor titular
Profesor tesorero	clas_id_profesor_teso		13			Profesor tesorero
Profesor Suplente 1	clas_id_profesor_sup1		13			Profesor Suplente 1
Profesor Suplente 2	clas_id_profesor_sup2		13			Profesor Suplente 2
Lugares Máximos	clas_lugares_maximos	Not null	3			Lugares Máximos
Lugares Ocupados	clas_lugares_ocupados	Not null	3			Lugares Ocupados
Fecha inicio	clas_fec_inicio		Date			Fecha inicio
Fecha fin	clas_fec_fin		Date			Fecha fin
Tipo	clas_tipo		2			Ordinario/Extraordinario
Estado	clas_estado		2			Activo/Inactivo

Figura 4.1.5.7. Diccionario de Datos Tabla Clase

Evaluación						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id alumno	eval_id_alumno	Null	13		Alumno	Id alumno
Id clase	eval_id_clase	Null	4		clase	Id clase
Id periodo	eval_id_periodo	Null	5		Periodo	Id periodo
Id grupo	eval_id_grupo	Null	4		Grupo	Id grupo
Id asignatura	eval_id_asignatura	Null	4		periodo	Id asignatura
Calificación 1	eval_calificacion1	Null	1			Calificación 1
Calificación 2	eval_calificacion2	Null	1			Calificación 2
Calificación 3	eval_calificacion3	Null	1			Calificación 3
Calificación 4	eval_calificacion4	Null	1			Calificación 4
Calificación 5	eval_calificacion5	Null	1			Calificación 5
Calificación 6	eval_calificacion6	Null	1			Calificación 6
Calificación 7	eval_calificacion7	Null	1			Calificación 7
Calificación 8	eval_calificacion8	Null	1			Calificación 8
Calificación 9	eval_calificacion9	Null	1			Calificación 9
Calificación 10	eval_calificacion10	Null	1			Calificación 10
Calificación 11	eval_calificacion11	Null	1			Calificación 11
Calificación 12	eval_calificacion12	Null	1			Calificación 12
Calificación 13	eval_calificacion13	Null	1			Calificación 13
Calificación 14	eval_calificacion14	Null	1			Calificación 14
Calificación 15	eval_calificacion15	Null	1			Calificación 15
Calificación 16	eval_calificacion16	Null	1			Calificación 16
Calificación 17	eval_calificacion17	Null	1			Calificación 17
Calificación 18	eval_calificacion18	Null	1			Calificación 18
Calificación 19	eval_calificacion19	Null	1			Calificación 19
Calificación 20	eval_calificacion20	Null	1			Calificación 20
Calificación 21	eval_calificacion21	Null	1			Calificación

						21
Calificación 22	eval_calificacion22	Null	1			Calificación 22
Calificación 23	eval_calificacion23	Null	1			Calificación 23
Calificación 24	eval_calificacion24	Null	1			Calificación 24
Fecha evaluación	eval_fec_evaluacion	Null	date			Fecha evaluación
Ip	eval_ip_maquina	Null	15			Ip
Estado	eval_estado	Null	1			Estado
Comentario	eval_comentario	Null	200			Comentario
Id profesor	prof_id_profesor	Null	4		profesor	Id profesor
Nombre profesor	prof_nombre_completo	Null	100			Nombre profesor

Figura 4.1.5.8. Diccionario de Datos Tabla Evaluación

Evaluación Folio						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id alumno	Alum_id_alumno	Null	13		Alumno	Id alumno
Id periodo	Peri_id_periodo	Null	6		Periodo	Id periodo
Id folio	Folio_id_evaluación	null	4		Folio	Id folio

Figura 4.1.5.9. Diccionario de Datos Tabla Evaluación Folio

Evaluación Reporte Factor						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id profesor	evarep_id_profesor	Null	4		Profesor	Id profesor
Nombre	evarep_nombre_completo	Null	100			Nombre

Id asignatura	evarep_id_asignatura	Null	4		Asignatura	Id asignatura
Id clase	evarep_id_clase	Null	4		Clase	Id clase
Id periodo	evarep_id_periodo	Null	5		Periodo	Id periodo
Id grupo	evarep_id_grupo	Null	4		Grupo	Id grupo
Promedio 1	evarep_e1	Null	Double			Promedio 1
Promedio 2	evarep_e2	Null	Double			Promedio 2
Promedio 3	evarep_e3	Null	Double			Promedio 3
Desviación Estándar 1	desv_e1	Null	Double			Desviación Estándar 1
Desviación Estándar 2	desv_e2	Null	Double			Desviación Estándar 2
Desviación Estándar 3	desv_e3	Null	Double			Desviación Estándar 3
Total alumnos	total_alumnos	Null	Integer			Total alumnos
Mínimo 1	min_e1	Null	Integer			Mínimo 1
Máximo 1	max_e1	Null	Integer			Máximo 1
Mínimo 2	min_e2	Null	Integer			Mínimo 2
Máximo 2	max_e2	Null	Integer			Máximo 2
Mínimo 3	min_e3	Null	Integer			Mínimo 3
Máximo 3	max_e3	Null	Integer			Máximo 3

Figura 4.1.5.10. Diccionario de Datos Tabla Evaluación Reporte Factor

Evaluación Reporte Reactivos						
Nombre	Acronimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Id profesor	evareactivo_id_profesor	Null	4		Profesor	Id profesor
Nombre Completo	evareactivo_nombre_completo	Null	100			Nombre Completo
Id asignatura	evareactivo_id_asignatura	Null	4		Asignatura	Id asignatura
Id clase	evareactivo_id_clase	Null	4		Clase	Id clase
Id periodo	evareactivo_id_periodo	Null	5		Periodo	Id periodo
Id grupo	evareactivo_id_grupo	Null	4		Grupo	Id grupo
Total alumnos	evareactivo_total_alumnos	Null	integer			Total alumnos

Reactivo 1	evareactivo_r1	Null	40			Reactivo 1
Reactivo 2	evareactivo_r2	Null	40			Reactivo 2
Reactivo 3	evareactivo_r3	Null	40			Reactivo 3
Reactivo 4	evareactivo_r4	Null	40			Reactivo 4
Reactivo 5	evareactivo_r5	Null	40			Reactivo 5
Reactivo 6	evareactivo_r6	Null	40			Reactivo 6
Reactivo 7	evareactivo_r7	Null	40			Reactivo 7
Reactivo 8	evareactivo_r8	Null	40			Reactivo 8
Reactivo 9	evareactivo_r9	Null	40			Reactivo 9
Reactivo 10	evareactivo_r10	Null	40			Reactivo 10
Reactivo 11	evareactivo_r11	Null	40			Reactivo 11
Reactivo 12	evareactivo_r12	Null	40			Reactivo 12
Reactivo 13	evareactivo_r13	Null	40			Reactivo 13
Reactivo 14	evareactivo_r14	Null	40			Reactivo 14
Reactivo 15	evareactivo_r15	Null	40			Reactivo 15
Reactivo 16	evareactivo_r16	Null	40			Reactivo 16
Reactivo 17	evareactivo_r17	Null	40			Reactivo 17
Reactivo 18	evareactivo_r18	Null	40			Reactivo 18
Reactivo 19	evareactivo_r19	Null	40			Reactivo 19
Reactivo 20	evareactivo_r20	Null	40			Reactivo 20
Reactivo 21	evareactivo_r21	Null	40			Reactivo 21
Reactivo 22	evareactivo_r22	Null	40			Reactivo 22
Reactivo 23	evareactivo_r23	Null	40			Reactivo 23
Reactivo 24	evareactivo_r24	Null	40			Reactivo 24

Figura 4.1.5.11. Diccionario de Datos Tabla Evaluación Reporte Reactivo

Grado						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
grad_grado_estud	Grado Estudios	Null	4			Grado Estudios
grad_titulo	Id grado	Null	4		Profesor	Id grado

Figura 4.1.5.12. Diccionario de Datos Tabla Grado

Inscripción						
Nombre	Acrónimo	Acepta Nulos	Tipo de Datos Longitud	Tipo de Llave	Tabla Relacionada	Descripción
Insc_id_clase	Id clase		6	Fk	clase	Id clase
Insc_id_periodo	Id periodo		6	Fk	Periodo	Id periodo
Insc_grupo	Id grupo	1	6	Fk	Grupo	Id grupo
Insc_asignatura	Id asignatura		6	Fk	Asignatura	Id asignatura
Insc_id_alumno	Id alumno		13	Fk	Alumno	Id alumno
Insc_id_acta	Id acta		5			Id acta
Insc_fec_inscripcion	Fecha de inscripción		10			Fecha
Insc_hora_inscripcion	Hora de inscripción		10			Hora
Insc_posicion_acta	Posicion acta	Null	10			Poscion acta
Insc_estado	Estado inscripción		2			Estado
Insc_tipo_movimiento	Tipo movimiento		2			Movimiento
Insc_ip_maquina	Ip maquina		20			Ip maquina

Figura 4.1.5.13. Diccionario de Datos Tabla Inscripción

4.2 Implementación de la Base de Datos

A partir de la versión 8.4 de PostgreSQL, la empresa EnterpriseDB ha puesto a disposición de la comunidad un instalador fácil. Este programa instalador está disponible para Linux, Windows y MacOS X y está pensado para hacerle la vida más fácil al usuario que quiere empezar a utilizar PostgreSQL e instalarlo de manera fácil y rápida. Este instalador provee la versión libre de la comunidad del servidor PostgreSQL, los programas clientes por defecto y el cliente gráfico PgAdminIII.

Vamos a explicar cómo sería el proceso de instalación de PostgreSQL con este instalador en Windows XP SP3. Lo primero que tenemos que hacer es descargar la versión del programa de instalación que corresponda con nuestro sistema desde la página: <http://www.enterprisedb.com/products-services-training/pgdownload> (utilizamos la versión Version 9.2.1)

Una vez descargado lo tendremos que ejecutar como administrador en nuestro sistema, usuario administrador' en Windows.

La primera pantalla que se nos muestra es la bienvenida al instalador de PostgreSQL. A partir de ahora tendremos que pulsar "Next" cada vez que queramos avanzar al siguiente paso. Vea figura 4.2.1.



Figura 4.2.1 Ventana de Bienvenida para la instalación de PostgreSQL

En el siguiente paso tendremos que definir el directorio donde vamos a instalar todos los programas que vienen con esta versión de PostgreSQL. En nuestro caso, utilizaremos el valor por defecto que el programa nos sugiere. Vea Figura 4.2.2.

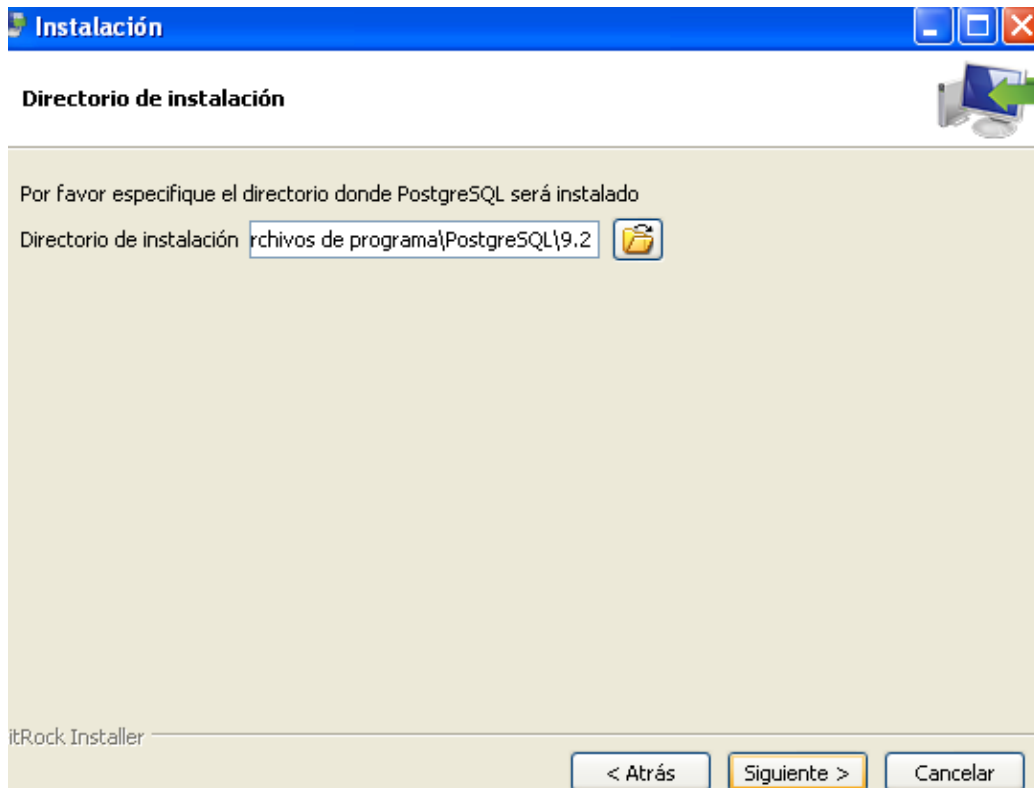


Figura 4.2.2 Directorio donde será instalado PostgreSQL

Una vez que hemos escrito el directorio que queremos utilizar para instalar el programa, pasamos al siguiente paso. En este paso tenemos que definir el directorio de datos en donde se crearán nuestras bases de datos. De nuevo, en nuestro caso utilizaremos el valor por defecto que el programa nos sugiere. Vea figura 4.3.2.

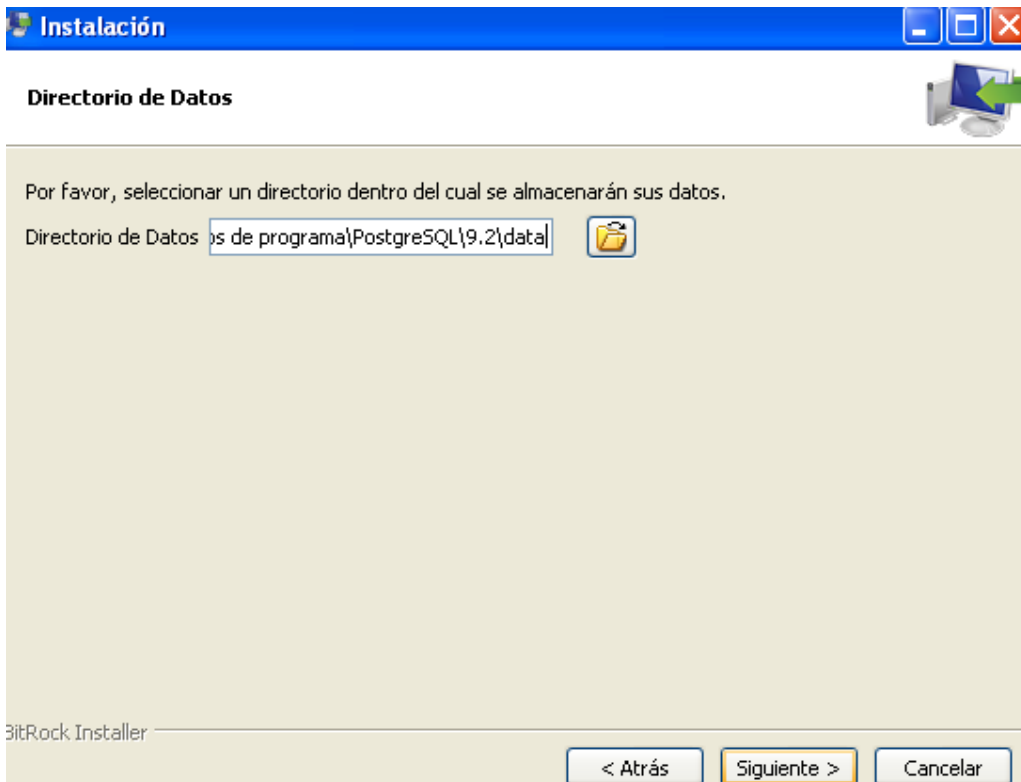


Figura 4.3.2 Ventana que indica la ruta donde se alojaran nuestros datos

Pasamos al siguiente paso. En este paso tenemos que definir una clave de acceso para el usuario administrador de nuestra base de datos PostgreSQL. Vea figura 4.2.4.

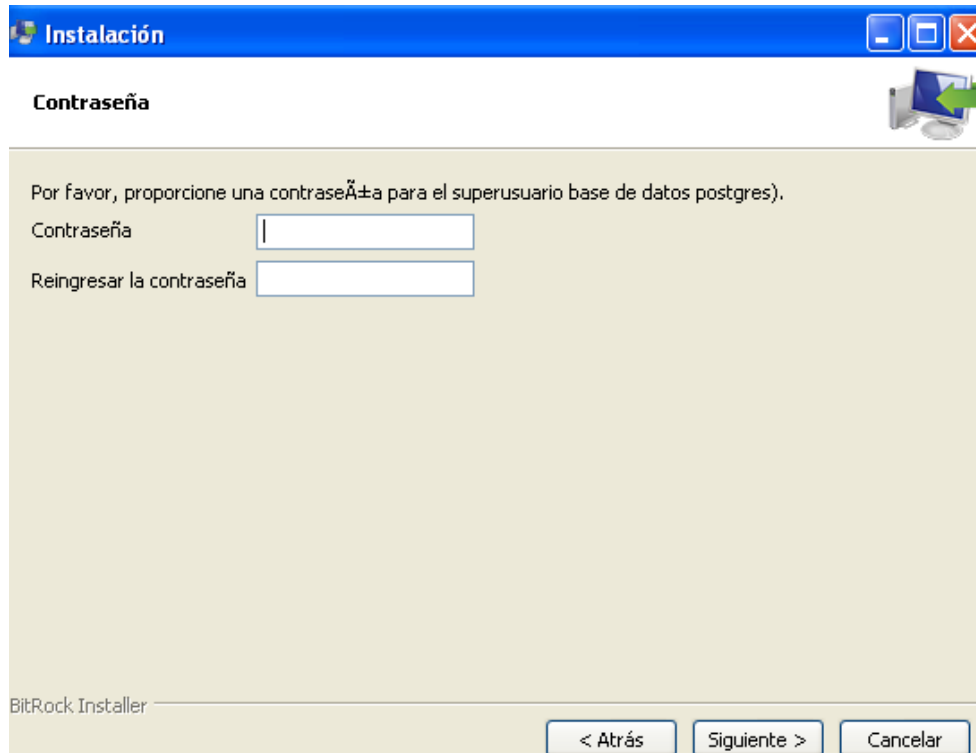


Figura 4.2.4 Contraseña para el superusuario

Pulsamos por última vez "Next" y esperamos a que el programa termine de instalar todo.

Una vez terminada la instalación, podremos salir del instalador pulsando "Finish". En este último paso el instalador nos da la opción de arrancar automáticamente un programa llamado "Stack Builder". Este programa puede ser usado para instalar diversos programas adicionales, tanto programas libres como algunos distribuidos por EnterpriseDB.

Creación de la Base de Datos

Usando el PgAdmin III el cual es una aplicación gráfica para el gestor de bases de datos PostgreSQL, La manera es la siguiente:

Colocamos el cursor en DataBase en la parte de objetos del navegador botón derecho y seleccionar New DataBase. Vea figura 4.2.5.

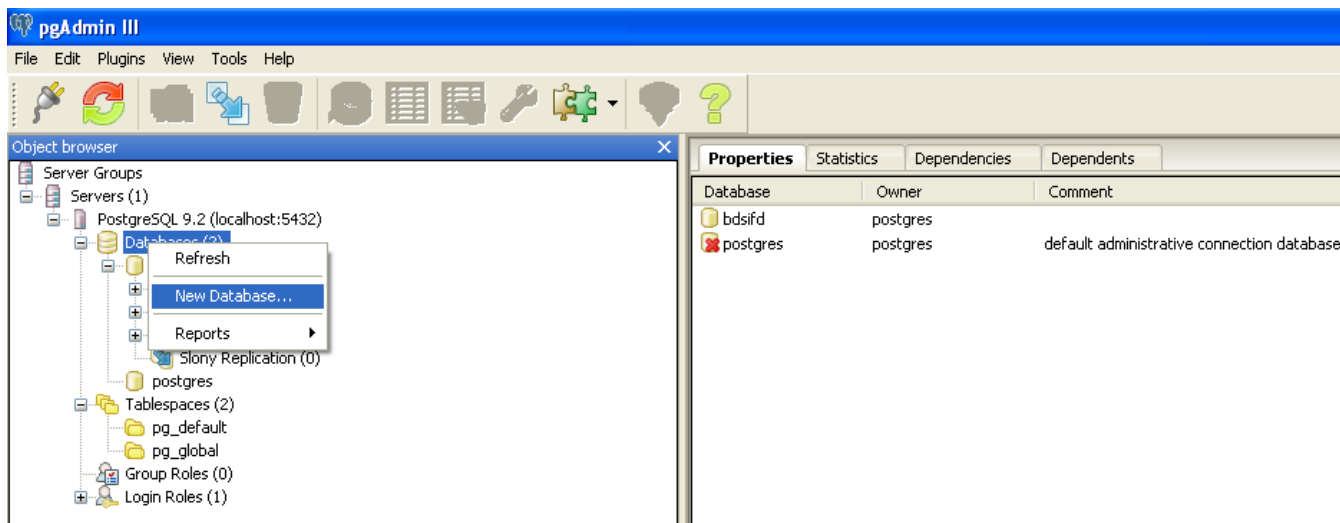


Figura 4.2.5 Creación de una base de datos

En propiedades proporcionamos el nombre de nuestra base datos y damos clic en ok. Nuestra base aparecerá dentro del árbol de DataBase, lista para usarse. Vea figura 4.2.6.

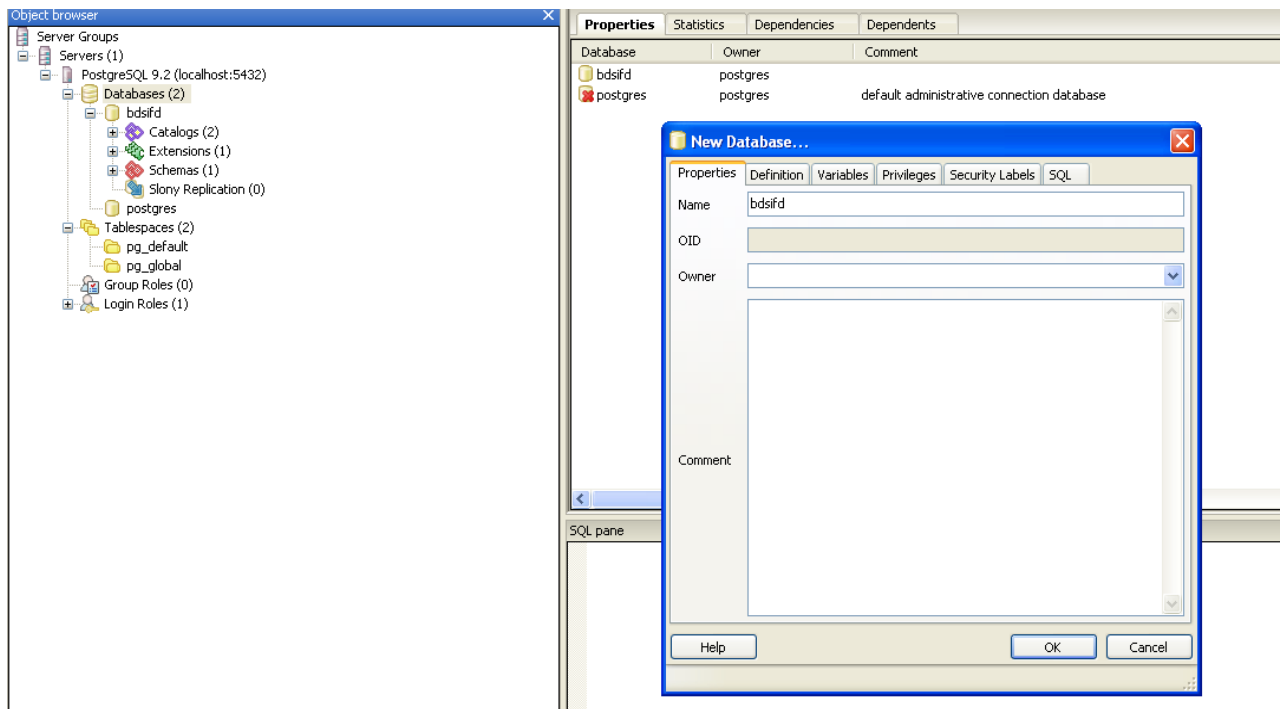


Figura 4.2.6 Propiedades de la base de datos

La sintaxis para crear una Base de Datos en SQL es la siguiente

```
CREATE DATABASE bdsifd
WITH ENCODING='UTF8'
CONNECTION LIMIT=-1;
CREATE DATABASE bdsifd
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
LC_COLLATE = 'Spanish_Mexico.1252'
```

```
LC_CTYPE = 'Spanish_Mexico.1252'
```

```
CONNECTION LIMIT = -1;
```

Crear tabla de datos

Para la creación de una tabla usando el PgAdmin III, seleccionamos la base de datos botón derecho, seleccionamos New Table, asignamos nombre a la tabla. Vea figura 4.2.7.

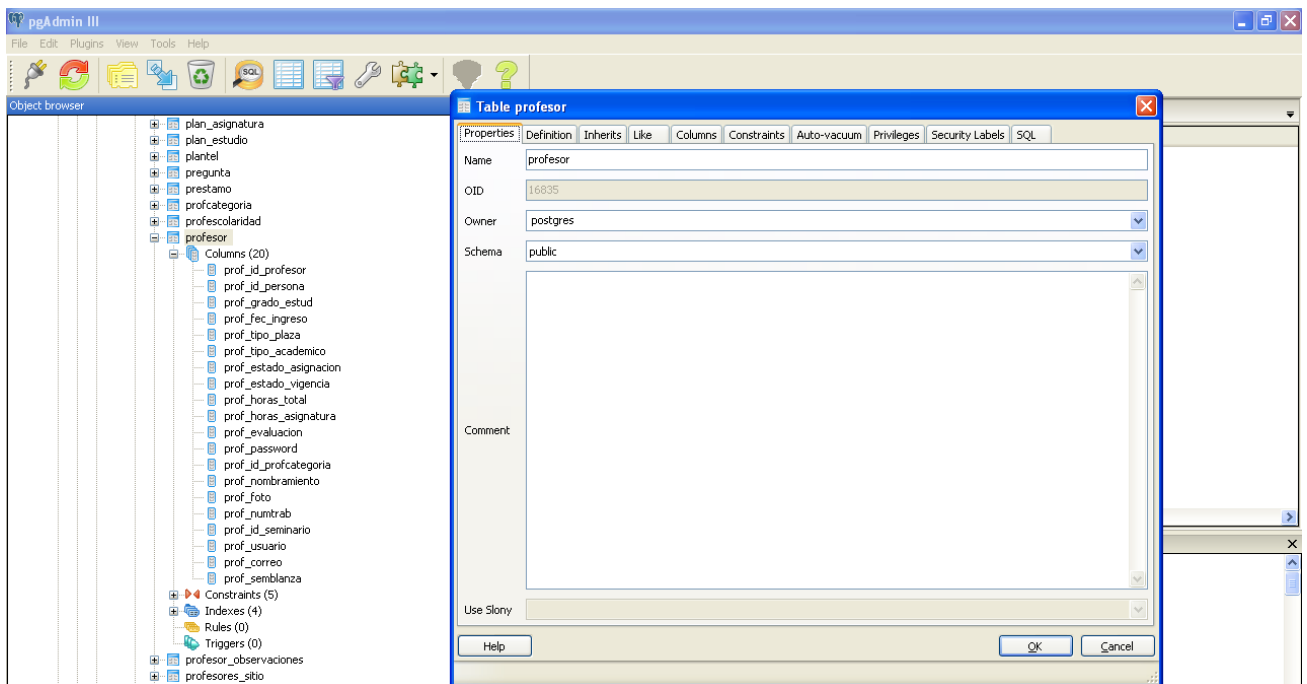


Figura 4.2.7 Propiedades de una tabla en PostgreSQL

Para agregar los campos en la tabla nos colocamos en la pestaña columnas donde deberemos asignar el nombre del campo, el tipo de dato que empleara y su longitud. Vea figura 4.2.8.

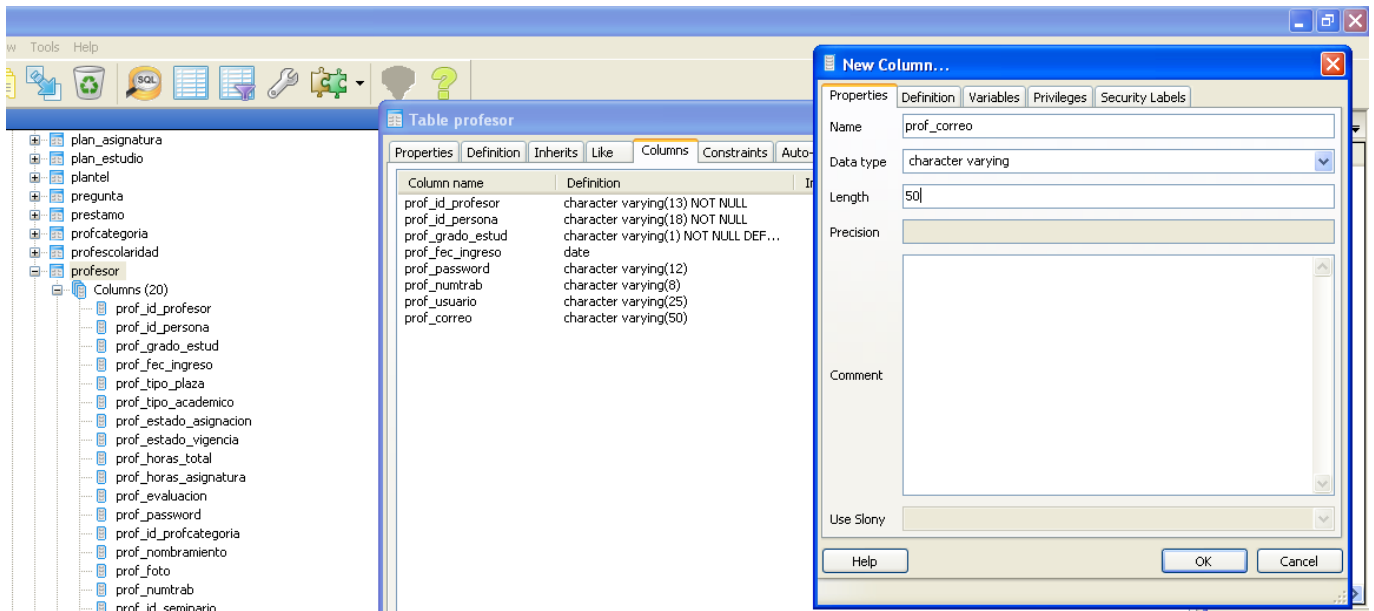


Figura 4.2.8 Creación de columnas dentro de la tabla de base de datos

La creación de una tabla usando SQL es de la siguiente manera:

CREATE TABLE profesor

(

prof_id_profesor character varying(13) NOT NULL,

prof_id_persona character varying(18) NOT NULL,

prof_grado_estud character varying(1) NOT NULL DEFAULT 0,

prof_fec_ingreso date,

prof_nivel character varying(1),

prof_tipo_plaza character varying(1),

prof_tipo_academico character varying(2) DEFAULT 'A'::character varying,

prof_estado_asignacion character varying(1),

```

prof_estado_vigencia character varying(2),
prof_horas_total character varying(2),
prof_horas_asignatura character varying(2),
prof_evaluacion character varying(7),
prof_password character varying(12),
prof_id_profcategoria character varying(2),
prof_nombramiento character varying(1),
prof_foto character varying(100),
prof_numtrab character varying(8),
prof_id_seminario character varying(3),
prof_usuario character varying(25) NOT NULL,
prof_correo character varying(50) NOT NULL,
prof_semlanza character varying(20),
CONSTRAINT profesor_pkey PRIMARY KEY (prof_id_profesor ),
CONSTRAINT profesor_prof_id_persona_fkey FOREIGN KEY (prof_id_persona)
REFERENCES persona (pers_id_persona) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT profesor_prof_id_profcategoria_fkey FOREIGN KEY
(prof_id_profcategoria)
REFERENCES profcategoria (pcat_id_profcategoria) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT uni_id_prof UNIQUE (prof_usuario ),
CONSTRAINT uni_prof_correo UNIQUE (prof_correo )

```

)

Árbol de tablas

Vea figuras 4.2.9 hasta la 4.2.21

Tabla Profesor

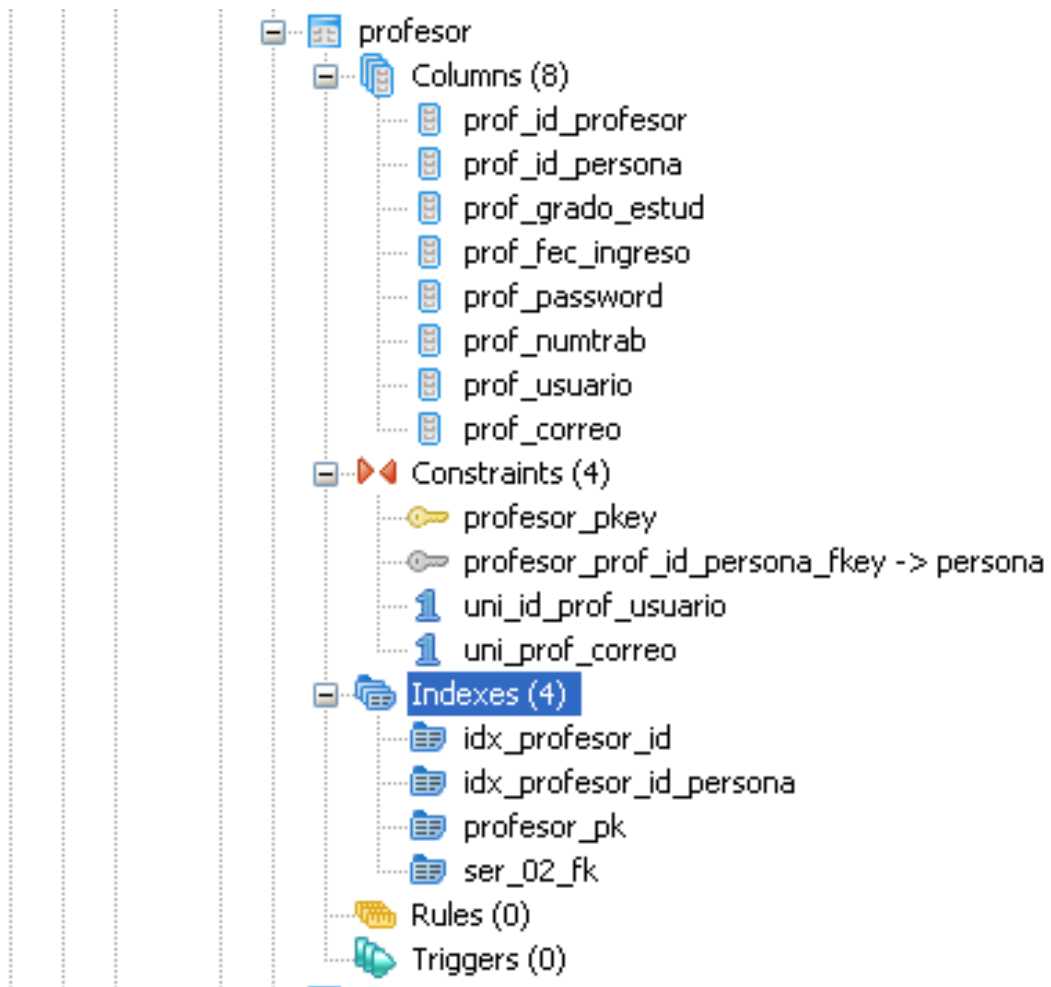


Figura 4.2.9 Árbol de tabla Profesor

Tabla Alumno

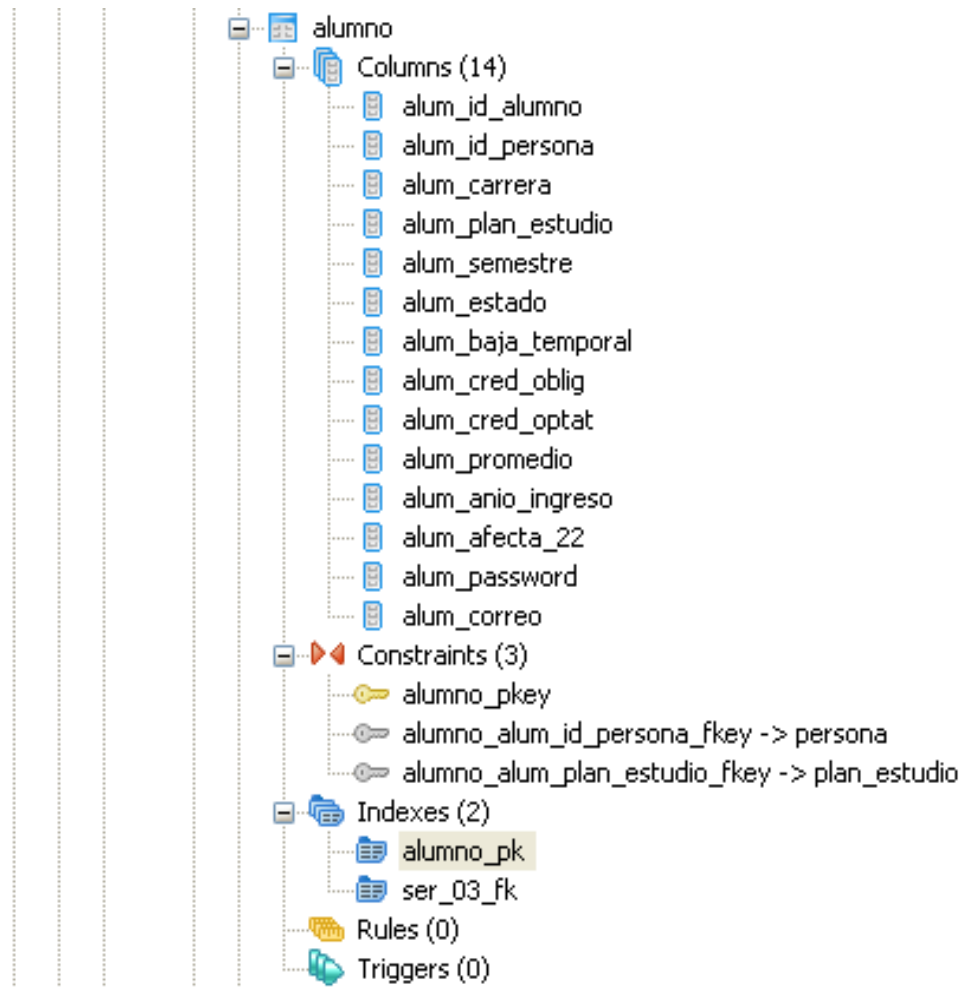


Figura 4.2.10 Árbol de tabla Alumno

Tabla Persona

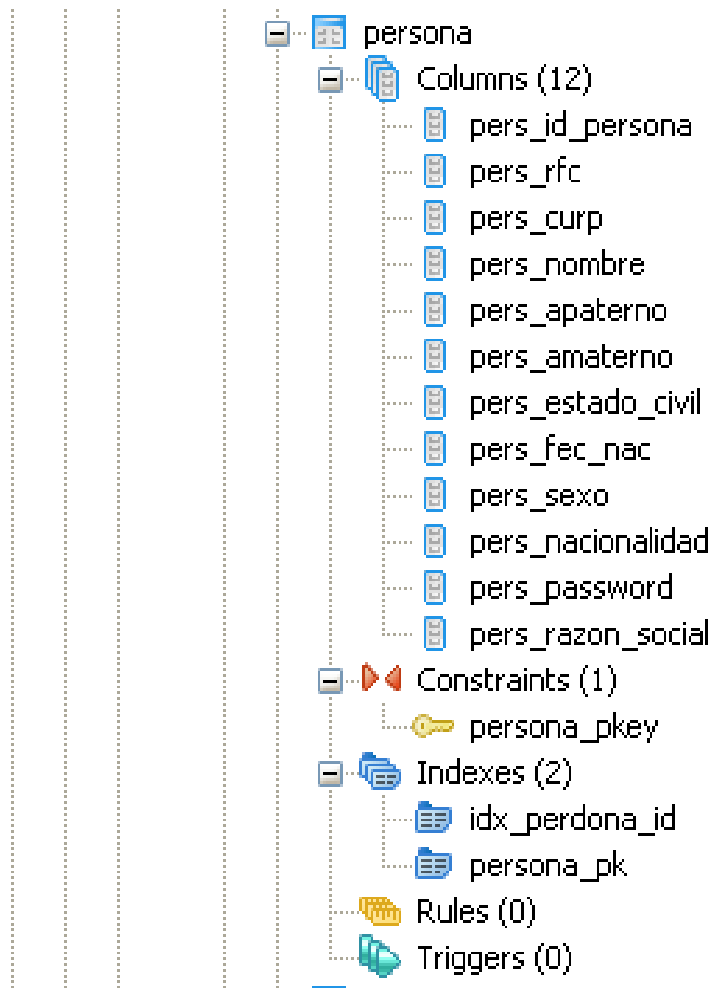


Figura 4.2.11 Árbol de tabla Persona

Tabla Periodo

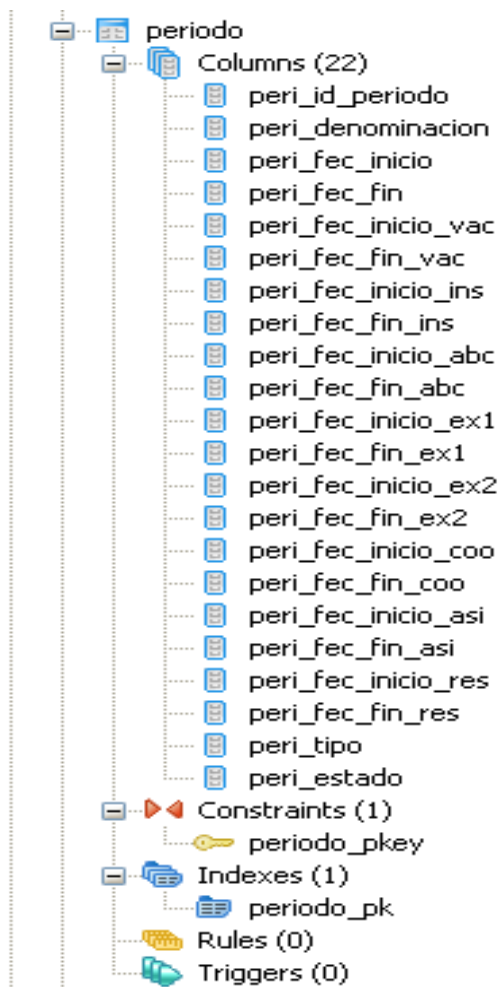


Figura 4.2.12 Árbol de tabla Periodo

Tabla Grado

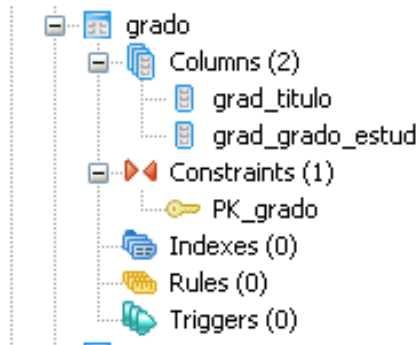


Tabla Clase



Figura 4.2.14 Árbol de tabla Clase

Tabla Grupo

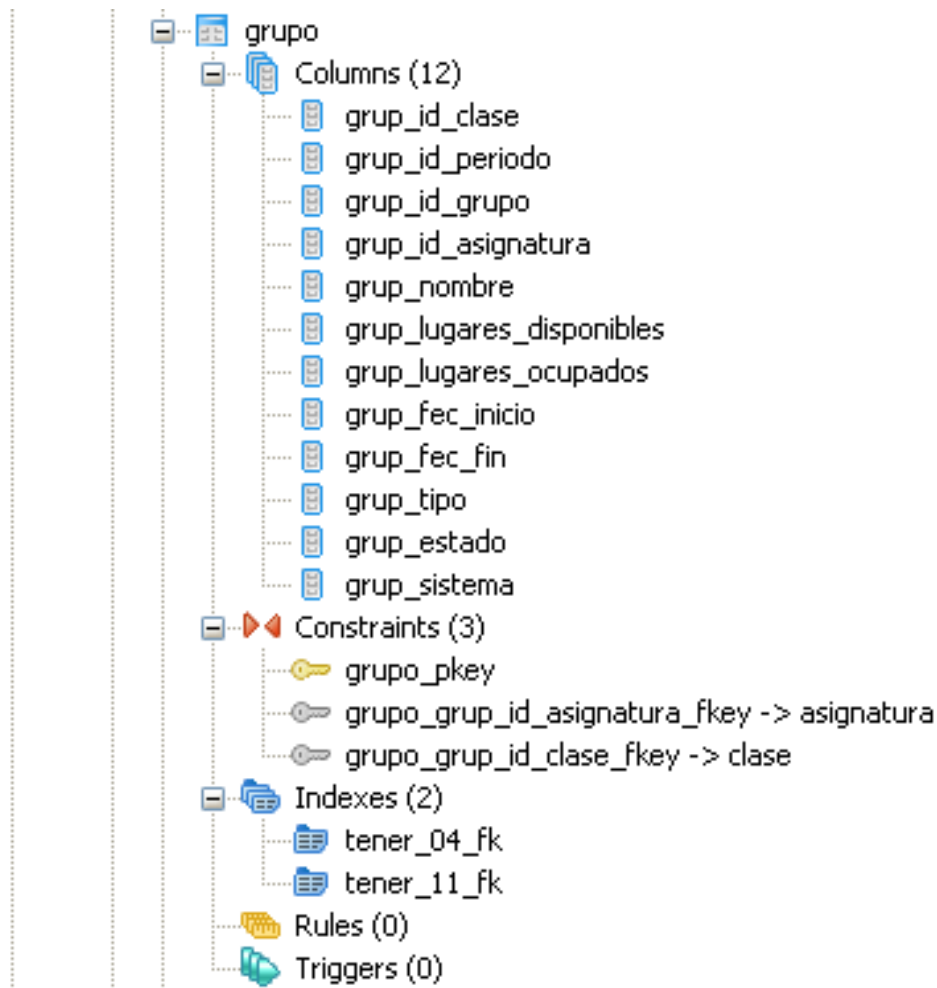


Figura 4.2.15 Árbol de tabla Grupo

Tabla Asignatura

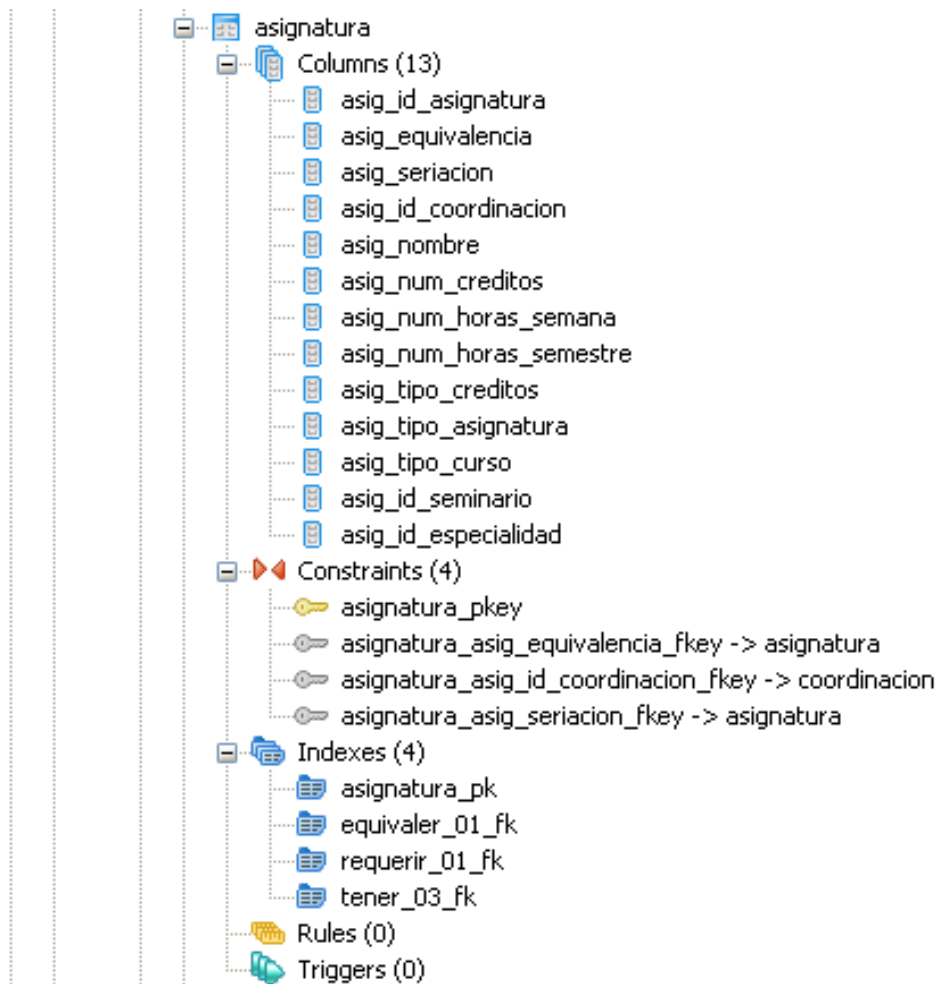


Figura 4.2.16 Árbol de tabla Asignatura

Tabla Inscripción

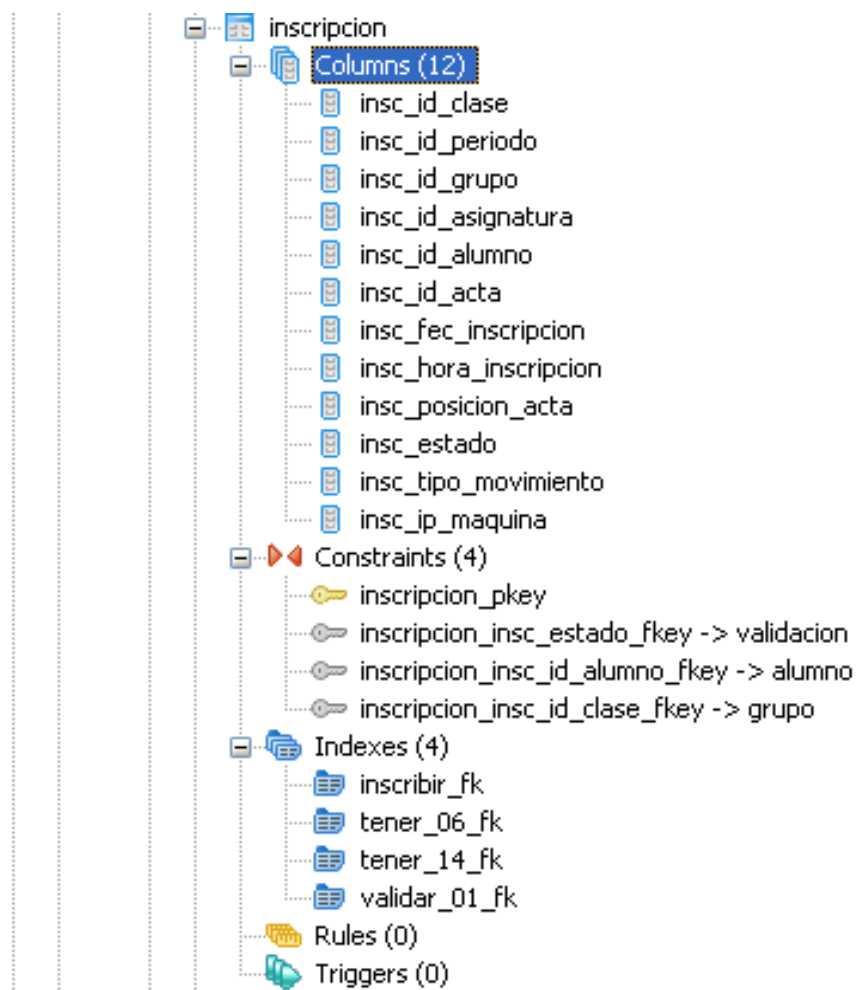


Figura 4.2.17 Árbol de tabla Inscripción

Tabla Evaluacion_reporte_factor



Figura 4.2.18 Árbol de tabla Evaluacion_reporte_factor

Tabla Evaluación Folio

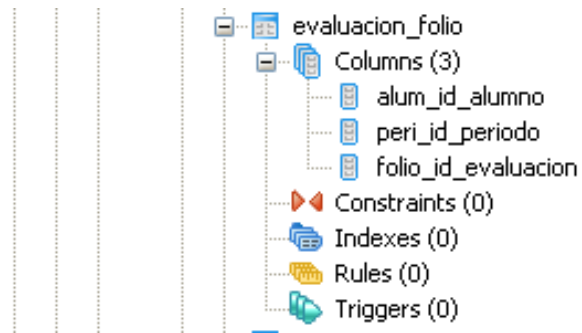


Figura 4.2.19 Árbol de tabla Evaluación folio

Tabla Evaluacion_reporte_reactivo



Figura 4.2.20 Arbol de tabla Evaluacion_reporte_reactivo

Tabla Evaluación

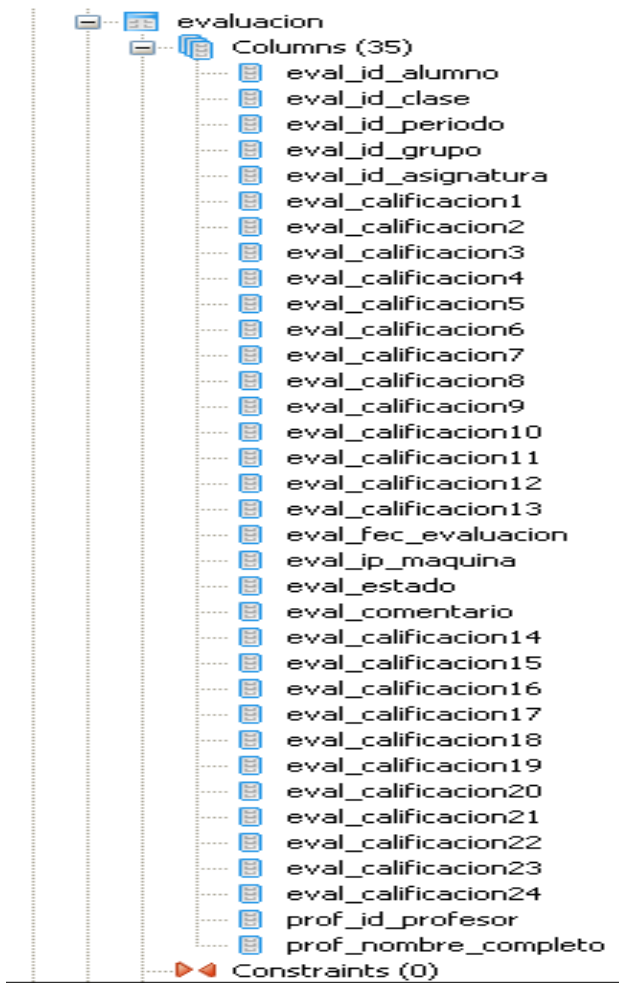


Figura 4.2.21 Árbol de tabla Evaluación

Consultas a través de SQL, separadas por asteriscos

```
SELECT alum_id_alumno, pers_nombre, pers_apaterno, pers_amaterno  
FROM persona, alumno WHERE alum_id_alumno = pers_id_persona  
AND alum_id_alumno = '$cta' and alum_password='$pwd'
```

```
SELECT * FROM evaluacion_folio  
WHERE alum_id_alumno='$txt_cta'  
AND peri_id_periodo='$periodo'
```

```
SELECT insc_id_grupo, insc_id_asignatura, insc_id_clase, insc_id_alumno,  
asig_nombre, prof_grado_estud, pers_nombre, pers_apaterno, pers_amaterno,  
insc_id_periodo, prof_id_profesor  
FROM INSCRIPCION, ASIGNATURA, GRUPO, CLASE, PROFESOR, PERSONA  
WHERE insc_id_periodo='$periodo'  
AND insc_id_clase = grup_id_clase  
AND insc_id_periodo = grup_id_periodo  
AND insc_id_asignatura = grup_id_asignatura  
AND insc_id_grupo = grup_id_grupo  
AND insc_id_asignatura = asig_id_asignatura  
AND insc_id_alumno = '$txt_cta'  
AND insc_tipo_movimiento = 'A'
```

```
AND clas_id_clase = insc_id_clase
AND clas_id_periodo = insc_id_periodo
    AND grup_tipo = '$tipgrup'
    AND clas_id_profesor_pres = prof_id_profesor
    AND prof_id_persona = pers_id_persona
    AND grup_estado != 'C' and insc_estado not like '02'
ORDER BY insc_id_asignatura, insc_id_grupo
```

```
*****
*****
```

```
SELECT eval_id_alumno FROM evaluacion_piloto_temp
    WHERE eval_id_alumno='$txt_cta'
    AND eval_id_grupo='$grupo'
    AND eval_id_asignatura='$cveasig'
    AND eval_id_periodo='$periodo'
```

```
insert into evaluacion_folio (alum_id_alumno, peri_id_periodo)
    values ('$txt_cta', '$periodo')
```

```
update evaluacion_folio set folio_id_evaluacion = '$folio'
    where alum_id_alumno='$txt_cta'
    AND peri_id_periodo='$periodo'
```

```
*****
*****
```

```
SELECT asig_nombre, prof_grado_estud, pers_nombre, pers_apaterno,  
pers_amaterno, prof_id_profesor
```

```
FROM INSCRIPCION, ASIGNATURA, GRUPO, CLASE, PROFESOR, PERSONA
```

```
WHERE insc_id_periodo='$periodo'
```

```
AND insc_id_clase = grup_id_clase
```

```
AND insc_id_clase = '$clase'
```

```
AND insc_id_periodo = grup_id_periodo
```

```
AND          insc_id_asignatura          =  
grup_id_asignatura
```

```
AND insc_id_grupo = grup_id_grupo
```

```
AND insc_id_grupo = '$grupo'
```

```
AND          insc_id_asignatura          =  
asig_id_asignatura
```

```
AND insc_id_asignatura = '$cveasig'
```

```
AND insc_id_alumno = '$txt_cta'
```

```
AND insc_tipo_movimiento = 'A'
```

```
AND clas_id_clase = insc_id_clase
```

```
AND clas_id_periodo = insc_id_periodo
```

```
AND grup_tipo = '$tipgrup'
```

```
AND          clas_id_profesor_pres      =  
prof_id_profesor
```

```
AND prof_id_persona = pers_id_persona
```

```
AND grup_estado != 'C' and insc_estado not like  
'02'
```

```
ORDER BY insc_id_asignatura, insc_id_grupo
```

```
SELECT eval_id_alumno from evaluacion_piloto_temp where

        eval_id_alumno = '$txt_cta' and

        eval_id_clase = '$clase' and

        eval_id_periodo = '$periodo' and

        eval_id_grupo = '$grupo' and

        eval_id_asignatura = '$cveasig'
```

```
INSERT INTO evaluacion_piloto_temp

(eval_id_alumno,eval_id_clase,eval_id_periodo,eval_id_grupo,

eval_id_asignatura,

eval_calificacion1,eval_calificacion2,eval_calificacion3,eval_calificacion4,

eval_calificacion5,eval_calificacion6,eval_calificacion7,eval_calificacion8,

eval_calificacion9, eval_calificacion10,eval_calificacion11,eval_calificacion12,

eval_calificacion13, eval_calificacion14, eval_calificacion15, eval_calificacion16,

eval_calificacion17, eval_calificacion18, eval_calificacion19, eval_calificacion20,

eval_calificacion21,eval_calificacion22, eval_calificacion23, eval_calificacion24,

eval_fec_evaluacion,eval_ip_maquina,eval_estado,eval_comentario,prof_id_profesor,

prof_nombre_completo)

VALUES ('$txt_cta','$clase','$periodo','$grupo','$cveasig',
```

```
'$preg1', '$preg2', '$preg3', '$preg4', '$preg5',  
'$preg6', '$preg7', '$preg8', '$preg9', '$preg10',  
'$preg11', '$preg12', '$preg13', '$preg14',  
'$preg15',  
'$preg16', '$preg17', '$preg18', '$preg19',  
'$preg20',  
'$preg21', '$preg22', '$preg23',  
$preg24', '$fecha', '$ip', 'C', '$comentario', '$idprofesor', '$nombreProf')
```

```
*****
```

```
*****
```

```
select * from list_eval_alum where lva_id_alumno='$cuenta'
```

```
*****
```

```
*****
```

```
INSERT INTO list_eval_alum VALUES ('$cuenta')
```

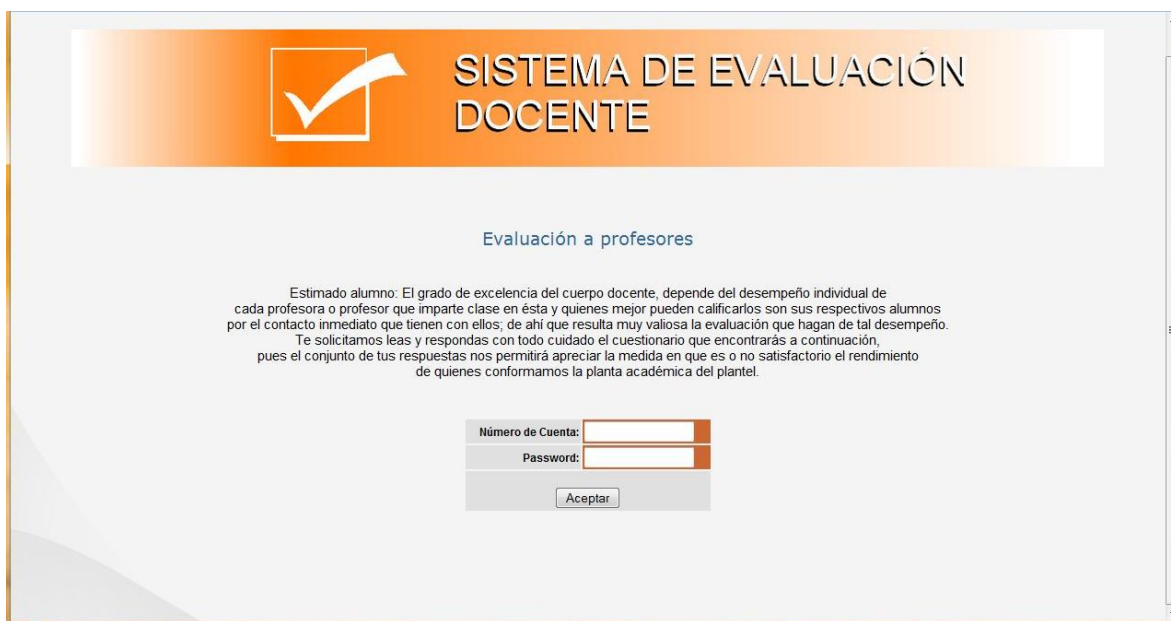
4.3 Diseño de Interfaz de usuario

El sistema de evaluación docente cuenta con un sitio principal para los alumnos. Toda la información requerida para el sistema será recopilada en una base de datos y de igual manera los datos que se generen a partir de las evaluaciones de los alumnos.

Se ha tenido en cuenta la seguridad de la información, ya que se tienen datos importantes tanto de las carreras y las asignaturas, como datos personales de alumnos y profesores.

Se ha elegido realizar el sistema con el lenguaje web PHP, éste permite el empleo de sesiones seguras para ingresar al mismo.

En la primera página, el sistema requiere la autenticación del usuario, como se muestra en la figura 4.3.1.



SISTEMA DE EVALUACIÓN DOCENTE

Evaluación a profesores

Estimado alumno: El grado de excelencia del cuerpo docente, depende del desempeño individual de cada profesora o profesor que imparte clase en ésta y quienes mejor pueden calificarlos son sus respectivos alumnos por el contacto inmediato que tienen con ellos; de ahí que resulta muy valiosa la evaluación que hagan de tal desempeño. Te solicitamos leas y respondas con todo cuidado el cuestionario que encontrarás a continuación, pues el conjunto de tus respuestas nos permitirá apreciar la medida en que es o no satisfactorio el rendimiento de quienes conformamos la planta académica del plantel.

Número de Cuenta:

Password:

Aceptar

Figura 4.3.1. Autenticación del usuario

Cualquier usuario, deberá validarse ante el sistema para poder ingresar a él y una vez validado, tendrá una sesión asignada.

Una vez que el usuario se ha identificado correctamente en el sistema, se mostrará una pantalla con las instrucciones para poder contestar el cuestionario de evaluación, en esta pantalla se muestra el botón de iniciar dando por entendido que se han leído las instrucciones y que va a comenzar el proceso de evaluación. (véase figura 4.3.2).

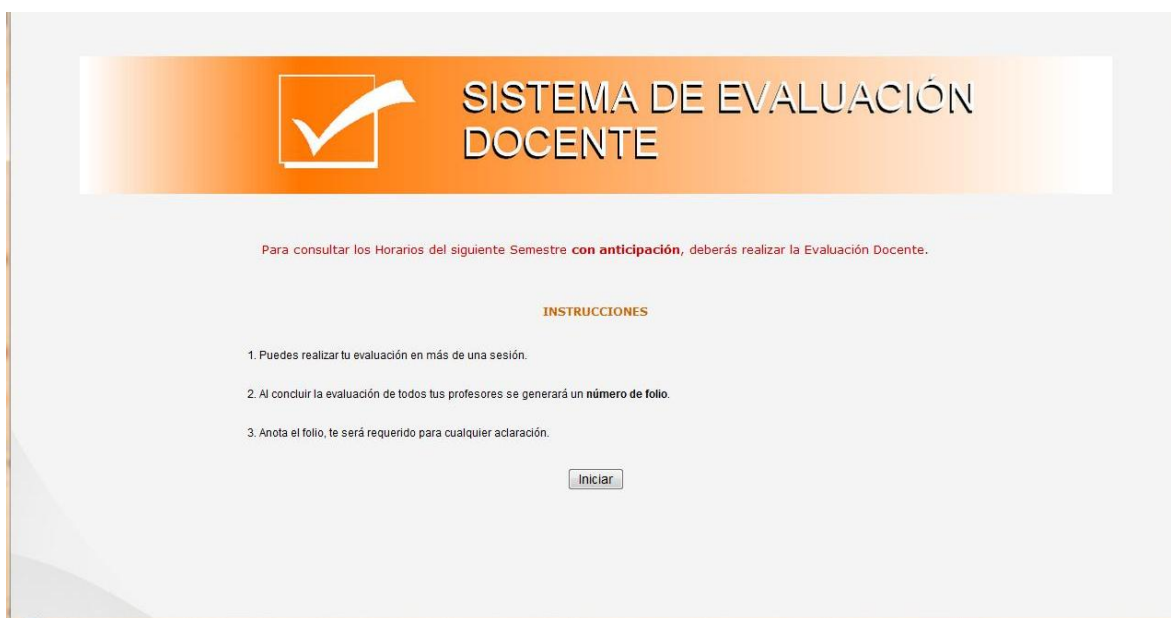


Figura 4.3.2. Instrucciones para iniciar la evaluación.

La siguiente pantalla mostrará la página para elegir la signatura a evaluar, así como el nombre del profesor que imparte esa asignatura, el módulo para elegir la asignatura trae la información de la base de datos de acuerdo al usuario que se ha identificado para conocer el semestre y las asignaturas en las que se encuentra inscrito (figura 4.3.3).

Para poder acceder a la evaluación, el usuario debe seleccionar la materia y dar click en el botón aceptar, el cual envía la información a la base de datos para

mostrar el cuestionario correspondiente, a su vez, si se desea elegir otra asignatura, se puede utilizar el botón de limpiar para cambiar a la opción deseada.

En esta pantalla también se muestra la opción de salir del sistema, este botón cierra la sesión del usuario. Se utiliza para poder continuar con otras asignaturas en otra sesión o para salir si se ha terminado la evaluación.

Alumno(a): ZAIRA ANAHI TORRES CRUZ			
EVALUACIÓN			
GRUPO	ASIGNATURA	PROFESOR	EVALUACIÓN COMPLETADA
0010	1513 - DERECHO PROCESAL PENAL	Lic. COSSIO ZAZUETA ARTURO LUIS	✓
0003	1616 - FAMILIA Y SUCESIONES	Lic. ARELLANO PALAFOX SARA	○
0003	1712 - AMPARO	Dr. SILVA RAMIREZ LUCIANO	✓
0008	1713 - DERECHO FISCAL II	Lic. VAZQUEZ ROBLES MIGUEL ANGEL RAFAEL	○
0009	1714 - DERECHO BANCARIO Y BURSATIL	Lic. MANZANILLA GALAVIZ JOSE MARIA	✓
0001	1715 - DERECHO INTERNACIONAL PRIVADO II	Dr. MANSILLA Y MEJIA MARIA ELENA	✓
0011	1716 - DERECHO COLECTIVO Y PROCESAL TRABAJO	Dr. RAMOS ALVAREZ OSCAR GABRIEL	✓
0005	1815 - FILOSOFIA DEL DERECHO	Dr. NUÑEZ CARPIZO ELSSIE	✓

ACEPTAR LIMPIAR SALIR DEL SISTEMA

Figura 4.3.3 Horario del alumno

Una vez que se eligió la asignatura deseada, el sistema podrá mostrar el cuestionario de evaluación.

En la siguiente pantalla se mostrará el cuestionario de evaluación que se extrae de la base de datos. La información sobre apertura de grupos, asignación de materias y grupos a profesores será obtenida de la información que provea la Administración Escolar. Estos datos solo serán gestionados por el administrador del sistema y la unidad de Servicios Escolares encargada para tal efecto. De esta manera, el administrador general podrá obtener reportes de cualquier carrera y cualquier profesor, sólo a manera de respaldo o cuando alguna autoridad lo solicite.

El cuestionario se divide en diferentes partes y en cada página se tiene el botón siguiente para poder continuar con cada parte de la evaluación. El cuestionario es de opción múltiple y está diseñado para que solo se pueda elegir una respuesta en cada pregunta. Las siguientes partes del cuestionario se muestran en las figuras 4.3.4, 4.3.5 y 4.3.6.

SISTEMA DE EVALUACIÓN DOCENTE

El conjunto de preguntas que aparecen a continuación tienen cinco opciones de respuesta. Selecciona la que refleje mejor tu experiencia con el profesor.

Profesor: ARELLANO PALAFOX SARA
Asignatura: FAMILIA Y SUCESIONES

PREGUNTAS	OPCIONES:				
	CASI NUNCA 0 - 19%	ALGUNAS VECES 20 - 39%	FRECUENTEMENTE 40 - 59%	CASI SIEMPRE 60 - 79%	SIEMPRE 80 - 100%
1. Respeta a los alumnos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
2. Domina los temas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
3. Permite el diálogo abierto con los alumnos.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Imparte su clase de forma dinámica.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Asiste puntualmente a impartir clase.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6. Vincula la teoría con la práctica.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7. Imparte su clase de manera clara.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
8. Evalúa el aprendizaje de manera objetiva.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

[Siguiente](#)

Figura 4.3.4 Inicio del cuestionario de evaluación.

SISTEMA DE EVALUACIÓN DOCENTE

Profesor: ARELLANO PALAFOX SARA
Asignatura: FAMILIA Y SUCESIONES

PREGUNTAS	OPCIONES:				
	CASI NUNCA 0 - 19%	ALGUNAS VECES 20 - 39%	FRECUENTEMENTE 40 - 59%	CASI SIEMPRE 60 - 79%	SIEMPRE 80 - 100%
9. Utiliza material didáctico o tecnológico para la enseñanza (lap top, cañón, diapositivas).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
10. Resuelve las dudas de los alumnos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
11. Enseña los valores de la profesión (ética y responsabilidad).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
12. Emplea diversas técnicas pedagógicas para impartir la clase.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13. Prepara su clase.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
14. Permite la participación de los alumnos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
15. Hace recomendaciones para que los alumnos tengan un buen desempeño en la carrera y en la vida profesional.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
16. Revisa los trabajos que deja.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Siguiete

Figura 4.3.5 Cuestionario de evaluación

SISTEMA DE EVALUACIÓN DOCENTE

Profesor: ARELLANO PALAFOX SARA
Asignatura: FAMILIA Y SUCESIONES

PREGUNTAS	OPCIONES:				
	CASI NUNCA 0 - 19%	ALGUNAS VECES 20 - 39%	FRECUENTEMENTE 40 - 59%	CASI SIEMPRE 60 - 79%	SIEMPRE 80 - 100%
17. Trata a los alumnos de manera equitativa.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
18. Enseña a respetar las opiniones de los compañeros de clase.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
19. Es ético (lleva a cabo lo que enseña).	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
20. Propone temas actuales para discutir en clase.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
21. Propone actividades extra clase para reforzar el aprendizaje.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
22. Exhorta a los alumnos a respetar el horario de clase.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
23. Enseña a los alumnos a comprender los temas.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
24. Realiza evaluaciones sólo de los temas revisados en clase.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Comentario: (200 caracteres)

Espacio para escribir algun comentario de tan solo 200 caracteres.

Terminar

Figura 4.3.6 Parte final del cuestionario.

En la última parte del cuestionario de evaluación se tiene un espacio de texto libre para poder ingresar los comentarios que se deseen. Este texto será guardado en la base de datos como texto plano.

Finalmente el sistema mostrará la confirmación si la evaluación se ha registrado con éxito una vez que se ha guardado la información obtenida en la base de datos del sistema (véase figura 4.3.7). A su vez mostrará el botón que permitirá al sistema regresar a la pantalla de elección de asignatura para poder continuar con las evaluaciones que falten o salir del sistema (figura 4.3.3).



Figura 4.3.7. Final de la evaluación.

El sistema de evaluación también cuenta con un pequeño modulo para la generación de los reportes. Véase figura 4.3.8.



Figura 4.3.8. Programa generador de reportes

El programa solo es para la utilización del administrador y con él podrá generar los reportes por factor y por reactivos en archivos PDF y guardarlos en su computadora. Véase figura 4.3.9.

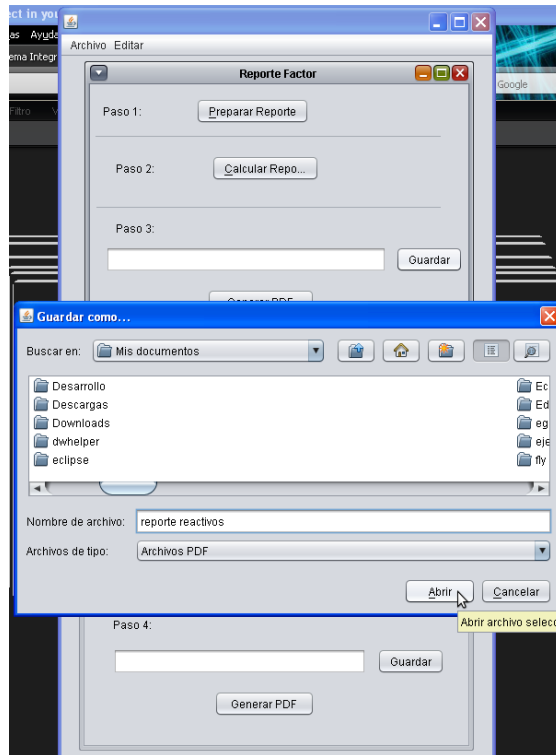


Figura 4.3.9. Generación de reportes

4.4 Generación de pruebas y mantenimiento

Las pruebas de software, son las investigaciones empíricas y técnicas cuyo fin es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada. Es una actividad más en el proceso de control de calidad.

Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo.

El objetivo de las pruebas es presentar información sobre la calidad del producto a las personas responsables de este.

Teniendo esta afirmación en mente, la información que puede ser requerida es de lo más variada. Esto hace que el proceso de testing sea completamente dependiente del contexto en el que se desarrolla.

A pesar de lo que muchos promueven, no existen las "mejores prácticas" como tal. Toda práctica puede ser ideal para una situación pero completamente inútil o incluso perjudicial en otra.

Por esto, las actividades, técnicas, documentación, enfoques y demás elementos que condicionaran las pruebas a realizar, deben ser seleccionadas y utilizadas de la manera más eficiente según el contexto del proyecto.

- **Pruebas estáticas**

Son el tipo de pruebas que se realizan sin ejecutar el código de la aplicación.

- **Pruebas dinámicas**

Todas aquellas pruebas que para su ejecución requieren la ejecución de la aplicación.

En las pruebas de software, conociendo la función específica para la que fue diseñado el producto, se pueden diseñar pruebas que demuestren que dicha función está bien realizada. Dichas pruebas son llevadas a cabo sobre la interfaz del software, es decir, de la función, actuando sobre ella como una caja negra, proporcionando unas entradas y estudiando las salidas para ver si concuerdan con las esperadas.

- **Pruebas de Caja Negra**

La prueba verifica que el ítem que se está probando, cuando se dan las entradas apropiadas produce los resultados esperados. Los datos de prueba se escogerán atendiendo a las especificaciones del problema, sin importar los detalles internos del programa, a fin de verificar que el programa corra bien.

El método de la caja negra se centra en los requisitos fundamentales del software y permite obtener entradas que prueben todos los requisitos funcionales del programa.

Con este tipo de pruebas se intenta encontrar:

Funciones incorrectas o ausentes.

- Errores de interfaz.
- Errores en estructuras de datos o en accesos a las bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y terminación.

Con la aplicación de esa técnica se obtiene un conjunto de pruebas que: Reduce el número de casos de pruebas y nos dicen algo sobre la presencia o ausencia de errores.

- Partición equivalente: Una partición equivalente es un método de prueba de caja negra que divide el dominio de entrada de un programa en clases de datos.
- Prueba de comparación: Esta técnica consiste en la comparación de salidas de un mismo software pero en sus diferentes versiones.

Algunos de los métodos empleados en las pruebas de caja negra son los siguientes:

- Métodos de prueba basados en grafos: en este método se debe entender los objetos (objetos de datos, objetos de programa tales como módulos o colecciones de sentencias del lenguaje de programación) que se modelan en el software y las relaciones que conectan a estos objetos. Una vez que se ha llevado a cabo esto, el siguiente paso es definir una serie de pruebas que verifiquen que todos los objetos tienen entre ellos las relaciones esperadas.
- Modelado del flujo de transacción. Los nodos representan los pasos de alguna transacción, y los enlaces representan las conexiones lógicas entre los pasos.
- Modelado de estado finito. Los nodos representan diferentes estados del software observables por el usuario, y los enlaces representan las transiciones que ocurren para moverse de estado a estado
- Modelado de flujo de datos. Los nodos objetos de datos y los enlaces son las transformaciones que ocurren para convertir un objeto de datos en otro.

- Modelado de planificación. Los nodos son objetos de programa y los enlaces son las conexiones secuenciales entre esos objetos. Los pesos de enlace se usan para especificar los tiempos de ejecución requeridos al ejecutarse el programa.

- **Pruebas de caja blanca**

Las pruebas de caja blanca (también conocidas como pruebas de caja de cristal o pruebas estructurales) se centran en los detalles procedurales del software, por lo que su diseño está fuertemente ligado al código fuente. El testeador escoge distintos valores de entrada para examinar cada uno de los posibles flujos de ejecución del programa y cerciorarse de que se devuelven los valores de salida adecuados.

Al estar basadas en una implementación concreta, si ésta se modifica, por regla general las pruebas también deberán rediseñarse.

Aunque las pruebas de caja blanca son aplicables a varios niveles —unidad, integración y sistema—, habitualmente se aplican a las unidades de software. Su cometido es comprobar los flujos de ejecución dentro de cada unidad (función, clase, módulo, etc.) pero también pueden testear los flujos entre unidades durante la integración, e incluso entre subsistemas, durante las pruebas de sistema.

A pesar de que este enfoque permite diseñar pruebas que cubran una amplia variedad de casos de prueba, podría pasar por alto partes incompletas de la especificación o requisitos faltantes, pese a garantizar la prueba exhaustiva de todos los flujos de ejecución del código analizado.

Las principales técnicas de diseño de pruebas de caja blanca son:

- Pruebas de flujo de control

- Pruebas de flujo de datos
- Pruebas de bifurcación (branch testing)
- Pruebas de caminos básicos

- **Pruebas de regresión**

Se denominan Pruebas de regresión a cualquier tipo de pruebas de software que intentan descubrir las causas de nuevos errores (bugs), carencias de funcionalidad, o divergencias funcionales con respecto al comportamiento esperado del software, inducidos por cambios recientemente realizados en partes de la aplicación que anteriormente al citado cambio no eran propensas a este tipo de error. Esto implica que el error tratado se reproduce como consecuencia inesperada del citado cambio en el programa.

Este tipo de cambio puede ser debido a prácticas no adecuadas de control de versiones, falta de consideración acerca del ámbito o contexto de producción final y extensibilidad del error que fue corregido (fragilidad de la corrección), o simplemente una consecuencia del rediseño de la aplicación.

Por lo tanto, en la mayoría de las situaciones del desarrollo de software se considera una buena práctica que cuando se localiza y corrige un bug, se grabe una prueba que exponga el bug y se vuelvan a probar regularmente después de los cambios subsiguientes que experimente el programa.

Existen herramientas de software que permiten detectar este tipo de errores de manera parcial o totalmente automatizada, la práctica habitual en programación extrema es que este tipo de pruebas se ejecuten en cada uno de los pasos del ciclo de vida del desarrollo del software.

Tipos de regresión

Clasificación de ámbito

- Local - los cambios introducen nuevos errores.
- Desenmascarada - los cambios revelan errores previos.
- Remota - Los cambios vinculan algunas partes del programa (módulo) e introducen errores en ella.

Clasificación temporal

- Nueva característica - los cambios realizados con respecto a nuevas funcionalidades en la versión introducen errores en otras novedades en la misma versión del software.
- Característica preexistente - los cambios realizados con respecto a nuevas funcionalidades introducen errores en funcionalidad existente de previas versiones.

Las Pruebas de Regresión pueden usarse no solo para probar la corrección de un programa, sino a menudo usarse para rastrear la calidad de su salida.

Las pruebas de regresión pueden incluir:

- La repetición de los casos de pruebas que se han realizado anteriormente y están directamente relacionados con la parte del sistema modificada.
- La revisión de los procedimientos manuales preparados antes del cambio, para asegurar que permanecen correctamente.
- La obtención impresa del diccionario de datos de forma que se compruebe que los elementos de datos que han sufrido algún cambio son correctos.
- **Pruebas de Integración.**

El objetivo de las pruebas de integración es verificar el correcto ensamblaje entre los distintos componentes una vez que han sido probados unitariamente con el fin de comprobar que interactúan correctamente a través de sus interfaces, tanto internas como externas, cubren la funcionalidad establecida y se ajustan a los requisitos no funcionales especificados en las verificaciones correspondientes.

Los tipos fundamentales de integración son los siguientes:

Integración incremental: se combina el siguiente componente que se debe probar con el conjunto de componentes que ya están probados y se va incrementando progresivamente el número de componentes a probar.

Integración no incremental: se prueba cada componente por separado y posteriormente se integran todos de una vez realizando las pruebas pertinentes.

- **Pruebas de validación**

Las pruebas de validación en la ingeniería de software son el proceso de revisión que el sistema de software producido cumple con las especificaciones y que cumple su cometido. Es normalmente una parte del proceso de pruebas de software de un proyecto, que también utiliza técnicas tales como evaluaciones, inspecciones, y tutoriales. La validación es el proceso de comprobar lo que se ha especificado es lo que el usuario realmente quería.

Se trata de evaluar el sistema o parte de este durante o al final del desarrollo para determinar si satisface los requisitos iniciales. La pregunta a realizarse es: ¿Es esto lo que el cliente quiere?

Enfoques a la verificación

- Dinámica de verificación, también conocido como ensayos o experimentación.
- Estática de verificación, también conocido como análisis.

Tipos

- Pruebas de aceptación: desarrolladas por el cliente.
- Pruebas alfa realizadas por el usuario con el desarrollador como observador en un entorno controlado (simulación de un entorno de producción).
- Pruebas beta: realizadas por el usuario en su entorno de trabajo y sin observadores.

Características

Comprobar que se satisfacen los requisitos:

- Se usan la mismas técnicas, pero con otro objetivo.
- No hay programas de prueba, sino sólo el código final de la aplicación.
- Se prueba el programa completo.
- Uno o varios casos de prueba por cada requisito o caso de uso especificado.
- Se prueba también rendimiento, capacidad, etc. (y no sólo resultados correctos).
- Pruebas alfa (desarrolladores) y beta (usuarios).

- **Pruebas alfa y beta**

Cuando se construye software a la medida para un cliente, se lleva a cabo una serie de pruebas de aceptación para permitir que el cliente valide todos los

requisitos. La mayoría de los desarrolladores de productos de software llevan a cabo un proceso denominado pruebas alfa y beta para descubrir errores que parezca que sólo el usuario final puede descubrir.

- **Prueba alfa:** se lleva a cabo, por un cliente, en el lugar de desarrollo. Se usa el software de forma natural con el desarrollador como observador del usuario y registrando los errores y problemas de uso. Las pruebas alfa se llevan a cabo en un entorno controlado.
- **Prueba beta:** se llevan a cabo por los usuarios finales del software en los lugares de trabajo de los clientes. A diferencia de la prueba alfa, el desarrollador no está presente normalmente. Así, la prueba beta es una aplicación en vivo del software en un entorno que no puede ser controlado por el desarrollador. El cliente registra todos los problemas que encuentra durante la prueba beta e informa a intervalos regulares al desarrollador.

A continuación se muestran las pruebas que se le realizaron al sistema:

En las siguientes pantallas visualizamos la ejecución completa del sistema y con ello realizamos las pruebas de Integración y Validación del sistema en donde se verifico el correcto ensamblaje entre los distintos componentes comprobando la interacción correcta a través de sus interfaces, cubren la funcionalidad establecida y cumplen con las especificaciones requeridas para el sistema de la evaluación docente en una institución educativa.

En la figura 4.4.1.1 y 4.4.1.2 se realizaron las pruebas de caja negra y de caja blanca ya que es necesarios que se valide el número de cuenta del alumno y su Password, estos datos deben cumplir la validación de que el alumno este dado de alta e inscrito en el semestre en el cual se realizara la encuesta, así como el Password sea el correcto.

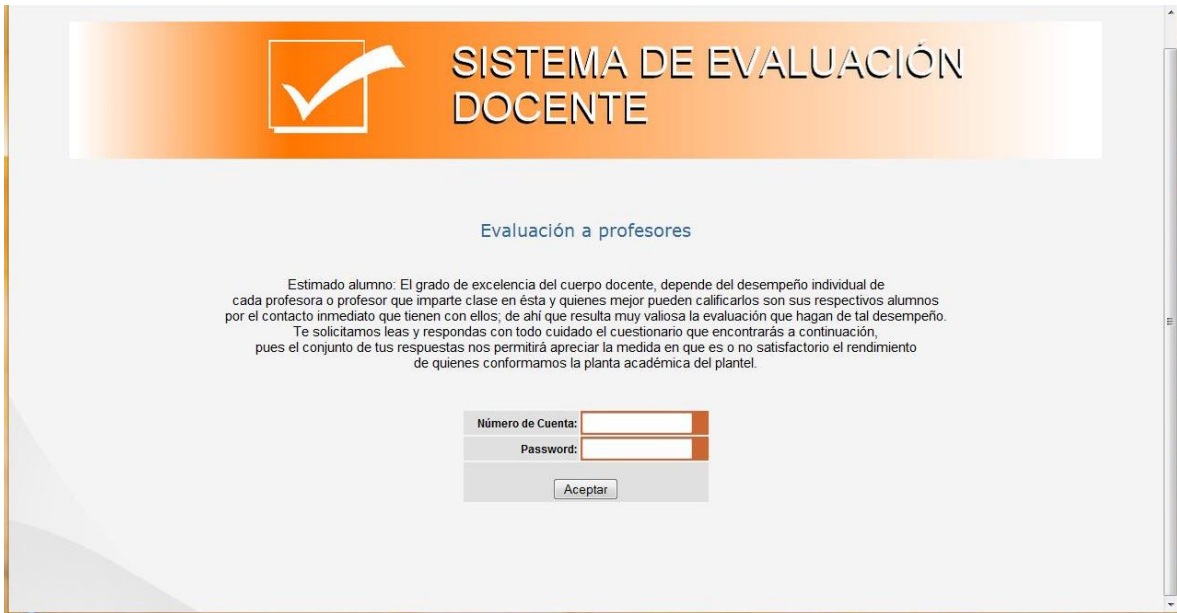


Figura 4.4.1.1 Validación de Entrada Prueba Caja Blanca

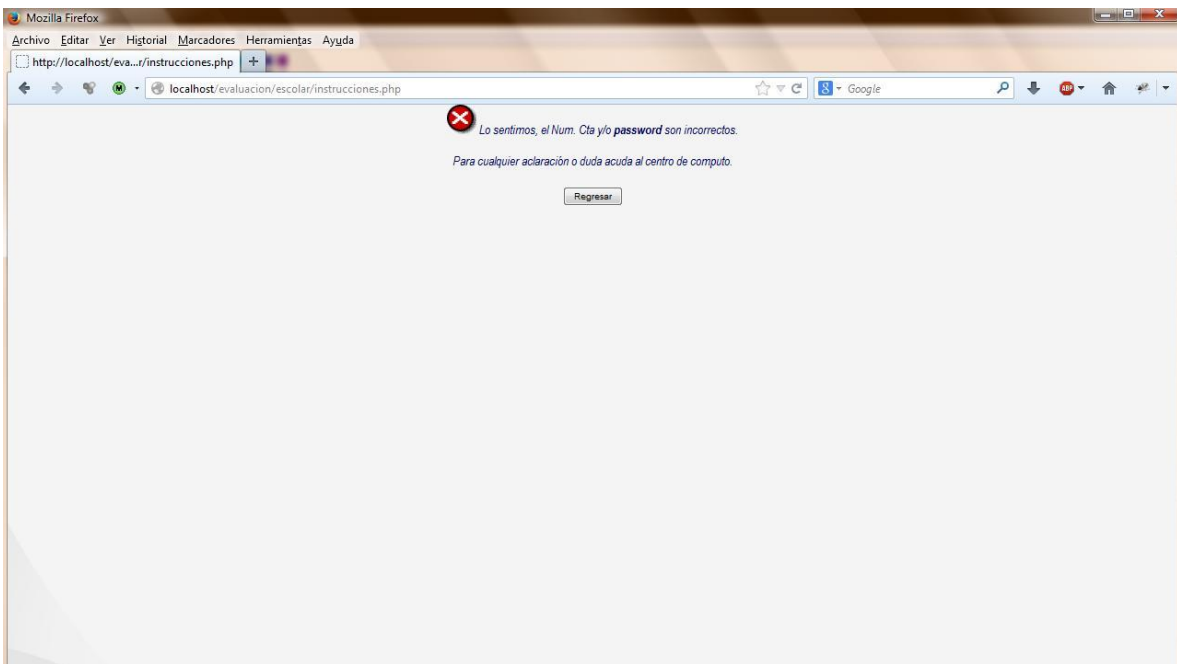


Figura 4.4.1.2 Validación de Entrada Prueba Caja Negra

En la Figura 4.4.2 se valida que la información ingresada por el alumno fue correcta para que se le puede presentar las Instrucciones para realizar la evaluación. Es en esta pantalla se realizo la prueba de Caja Negra.



Figura 4.4.2 Validación de Estatus de evaluación

En la Figura 4.4.3 se valida el estatus de la evaluación, en donde se le presente al alumno aquellas encuestas de evaluaciones que ha realizado y cuales están pendientes de realizar, se realiza la validación de la Asignatura y profesor sean los correctos en los cuales el alumno esta inscrito y el profesor sea el que imparte la asignatura correspondiente. Aquí se puede observar las pruebas Alfa y Beta las cuales fueron realizadas por el usuario, así como las pruebas de caja negra y caja blanca.

Alumno(a): ZAIRA ANAHI TORRES CRUZ

GRUPO	ASIGNATURA	PROFESOR	EVALUACIÓN COMPLETADA
0010	1513 - DERECHO PROCESAL PENAL	Lic. COSSIO ZAZUETA ARTURO LUIS	<input checked="" type="checkbox"/>
0003	1616 - FAMILIA Y SUCESIONES	Lic. ARELLANO PALAFOX SARA	<input checked="" type="checkbox"/>
0003	1712 - AMPARO	Dr. SILVA RAMIREZ LUCIANO	<input checked="" type="checkbox"/>
0008	1713 - DERECHO FISCAL II	Lic. VAZQUEZ ROBLES MIGUEL ANGEL RAFAEL	<input checked="" type="checkbox"/>
0009	1714 - DERECHO BANCARIO Y BURSÁTIL	Lic. MANZANILLA GALAVIZ JOSE MARIA	<input checked="" type="checkbox"/>
0001	1715 - DERECHO INTERNACIONAL PRIVADO II	Dr. MANSILLA Y MEJIA MARIA ELENA	<input checked="" type="checkbox"/>
0011	1716 - DERECHO COLECTIVO Y PROCESAL TRABAJO	Dr. RAMOS ALVAREZ OSCAR GABRIEL	<input checked="" type="checkbox"/>
0005	1815 - FILOSOFIA DEL DERECHO	Dr. NUÑEZ CARPIZO ELISSE	<input checked="" type="checkbox"/>

ACEPTAR LIMPIAR SALIR DEL SISTEMA

Figura 4.4.3 Validación de llenado de encuesta

En las figura 4.4.4, 4.4.5, 4.4.6.1 y 4.4.6.2 son donde se valida el correcto llenado de la encuesta de evaluación docente, puesto que el alumno tiene que elegir alguna de las opciones para cada pregunta para que pueda proseguir con la encuesta. Aquí realizamos la prueba de caja blanca y caja negra.

El conjunto de preguntas que aparecen a continuación tienen cinco opciones de respuesta. Selecciona la que refleje mejor tu experiencia con el profesor.

Profesor: ARELLANO PALAFOX SARA
Asignatura: FAMILIA Y SUCESIONES

PREGUNTAS:	OPCIONES:				
	CASI NUNCA 0 - 19%	ALGUNAS VECES 20 - 39%	FRECUENTEMENTE 40 - 59%	CASI SIEMPRE 60 - 79%	SIEMPRE 80 - 100%
1. Respeta a los alumnos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
2. Domina los temas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
3. Permite el diálogo abierto con los alumnos.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Imparte su clase de forma dinámica.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Asiste puntualmente a impartir clase.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6. Vincula la teoría con la práctica.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7. Imparte su clase de manera clara.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
8. Evalúa el aprendizaje de manera objetiva.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

Siguiente

Figura 4.4.4 Validación de llenado de encuesta

SISTEMA DE EVALUACIÓN DOCENTE

Profesor: ARELLANO PALAFOX SARA
Asignatura: FAMILIA Y SUCESIONES

PREGUNTAS:	OPCIONES:				
	CASI NUNCA 0 - 19%	ALGUNAS VECES 20 - 39%	FRECUENTEMENTE 40 - 59%	CASI SIEMPRE 60 - 79%	SIEMPRE 80 - 100%
9. Utiliza material didáctico o tecnológico para la enseñanza (lap top, cañón, diapositivas).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
10. Resuelve las dudas de los alumnos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
11. Enseña los valores de la profesión (ética y responsabilidad).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
12. Emplea diversas técnicas pedagógicas para impartir la clase.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13. Prepara su clase.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
14. Permite la participación de los alumnos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
15. Hace recomendaciones para que los alumnos tengan un buen desempeño en la carrera y en la vida profesional.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
16. Revisa los trabajos que deja.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Siguiente

Figura 4.4.5 Validación de llenado de encuesta

SISTEMA DE EVALUACIÓN DOCENTE

Profesor: ARELLANO PALAFOX SARA
Asignatura: FAMILIA Y SUCESIONES

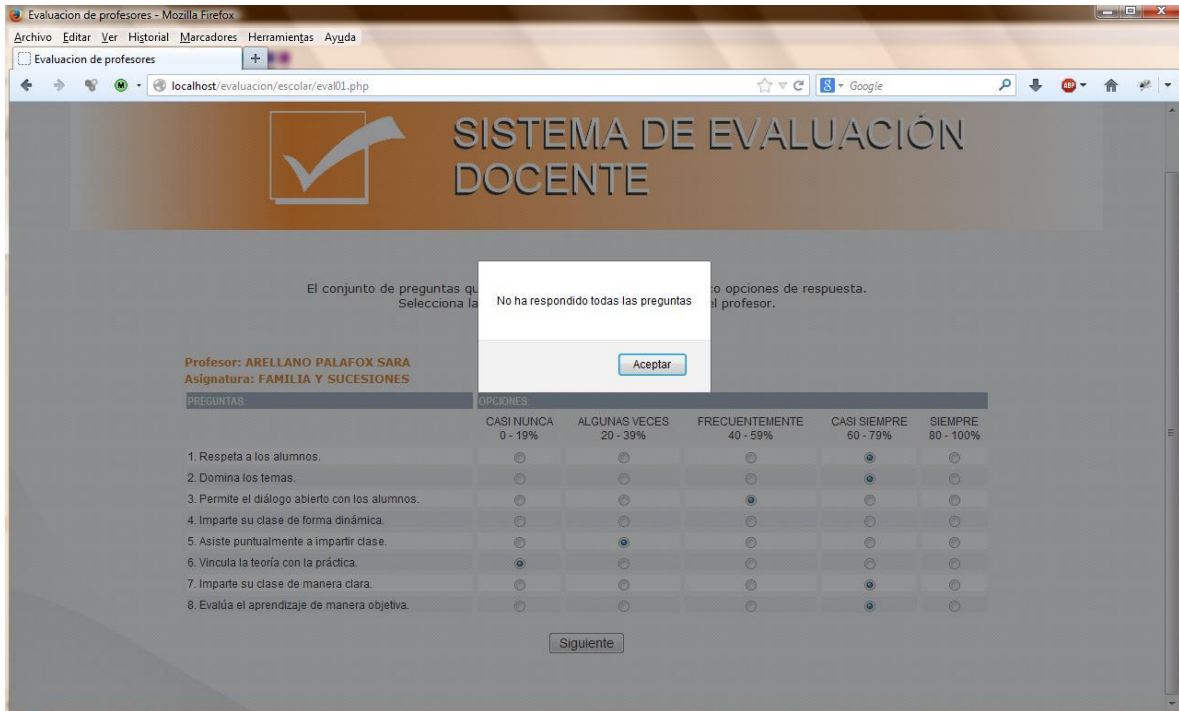
PREGUNTAS:	OPCIONES:				
	CASI NUNCA 0 - 19%	ALGUNAS VECES 20 - 39%	FRECUENTEMENTE 40 - 59%	CASI SIEMPRE 60 - 79%	SIEMPRE 80 - 100%
17. Trata a los alumnos de manera equitativa.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
18. Enseña a respetar las opiniones de los compañeros de clase.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
19. Es ético (lleva a cabo lo que enseña).	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
20. Propone temas actuales para discutir en clase.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
21. Propone actividades extra clase para reforzar el aprendizaje.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
22. Exhorta a los alumnos a respetar el horario de clase.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
23. Enseña a los alumnos a comprender los temas.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
24. Realiza evaluaciones sólo de los temas revisados en clase.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Comentario: (200 caracteres)

Espacio para escribir algun comentario de tan solo 200 caracteres.

Terminar

PANTALLA 4.4.6.1 Validación de información correcta Prueba Caja Blanca



PANTALLA 4.4.6.2 Validación de información correcta Prueba Caja Negra

En la Pantalla 4.4.7 se valida que la información ingresada por el alumno fue correcta y que todas las preguntas de la encuesta fueron contestadas el comentario es opcional, con esto se le presentara el mensaje 'Tu evaluación se ha registrado con éxito'. Es en esta pantalla se aplico la prueba de Caja Negra.



Figura 4.4.7 Validación de datos del alumno

En la Figura 4.4.8 se realizaron las validaciones del semestre correspondiente a la encuesta, el nombre del alumno que realizo la encuesta, las asignaturas en la cuales está inscrito el alumno así como los correspondientes profesores, se actualiza el estatus de la encuesta realizadas, una vez que estos datos son correctos se genera un número Folio para el estudiante, con esto se le presentara el mensaje ‘Concluiste con éxito la evaluación a tus profesores’. Es con esto que se da por terminada la evaluación docente. En esta parte se realiza la prueba de Integración y validación del sistema.

DOCENTE

Alumno(a): **ZAIRA ANAHI TORRES CRUZ**

GRUPO	ASIGNATURA	PROFESOR	EVALUACIÓN COMPLETADA
0010	1513 - DERECHO PROCESAL PENAL	Lic. COSSIO ZAZUETA ARTURO LUIS	✓
0003	1816 - FAMILIA Y SUCESIONES	Lic. ARELLANO PALAFOX SARA	✓
0003	1712 - AMPARO	Dr. SILVA RAMIREZ LUCIANO	✓
0008	1713 - DERECHO FISCAL II	Lic. VAZQUEZ ROBLES MIGUEL ANGEL RAFAEL	✓
0009	1714 - DERECHO BANCARIO Y BURSATIL	Lic. MANZANILLA GALAVIZ JOSE MARIA	✓
0001	1715 - DERECHO INTERNACIONAL PRIVADO II	Dr. MANSILLA Y MEJIA MARIA ELENA	✓
0011	1716 - DERECHO COLECTIVO Y PROCESAL TRABAJO	Dr. RAMOS ALVAREZ OSCAR GABRIEL	✓
0005	1815 - FILOSOFIA DEL DERECHO	Dr. NUÑEZ CARPIZO ELSIE	✓

Concluiste con éxito la evaluación de todos tus profesores.

Folio: 0037

No olvides anotar.

Gracias por participar.

Salir del Sistema

Figura 4.4.8 Validación de Término de encuesta

Mantenimiento de software

Es una de las actividades más comunes en la ingeniería de software, es el proceso de mejora y optimización del software después de su entrega al usuario final (es decir; revisión del programa), así como también corrección y prevención de los defectos.

El mantenimiento de software es también una de las fases en el ciclo de vida de desarrollo de sistemas (SDLC, sigla en inglés de system development life cycle), que se aplica al desarrollo de software. La fase de mantenimiento es la fase que viene después del despliegue (implementación) del software en el campo.

Fase de mantenimiento

La fase de mantenimiento de software involucra cambios al software para corregir defectos encontrados durante su uso o la adición de nueva funcionalidad mejorando la usabilidad y aplicabilidad del software.

La Mantenibilidad es la Capacidad de un producto de software de ser modificado. Estas modificaciones incluyen correcciones, mejoras, o adaptaciones a cambios en el entorno, los requisitos o las especificaciones funcionales

Tipos de Mantenimiento:

Existen 4 tipos de mantenimiento:

- Correctivo.
- Adaptativo.
- Perfectivo.
- Preventivo.

- **Mantenimiento correctivo:**

Tiene por objetivo localizar y eliminar los posibles defectos de los programas.

Un defecto en un sistema es una característica del sistema con el potencial de provocar un fallo. Un fallo se produce cuando el comportamiento de un sistema difiere con respecto al comportamiento definido en la especificación.

Los fallos en un sistema de software pueden ser:

- Procesamiento (salidas incorrectas de un programa).
- Rendimiento (tiempo de respuesta demasiado alto).
- Programación (inconsistencias en el diseño).
- Documentación (inconsistencias entre la funcionalidad de un programa y el manual de usuario).

- **Mantenimiento adaptativo:**

Consiste en la modificación de un programa debido a cambios en el entorno (hardware o software) en el que se ejecuta. Desde cambios en el sistema operativo, pasando por cambios en la arquitectura física del sistema informático, hasta en el entorno de desarrollo del software. Este tipo de mantenimiento puede ser desde un pequeño retoque hasta una reescritura de todo el código.

Los cambios en el entorno de desarrollo del software pueden ser:

- En el entorno de los datos (p.e. cambiar sistema de ficheros por BD relacional).
- En el entorno de los procesos (p.e. migración a plataforma con procesos distribuidos).

Este mantenimiento es cada vez más frecuente debido a la tendencia actual de actualización de hardware y Software cada poco tiempo.

- **Mantenimiento perfectivo:**

Conjunto de actividades para mejorar o añadir nuevas funcionalidades requeridas por el usuario.

Se divide en dos:

- Mantenimiento de Ampliación: incorporación de nuevas funcionalidades.
- Mantenimiento de Eficiencia: mejora de la eficiencia de ejecución.

- **Mantenimiento preventivo:**

Modificación del software para mejorar las propiedades de dicho software (calidad y mantenibilidad) sin alterar sus especificaciones funcionales. Incluir sentencias que comprueben la validez de los datos de entrada, reestructuración de los programas para aumentar su legibilidad o incluir nuevos comentarios. Este tipo de mantenimiento utiliza las técnicas de ingeniería inversa y reingeniería. El mantenimiento para la reutilización especializado en mejorar la reusabilidad del software se incluye en este tipo.

- **ACTIVIDADES DE MANTENIMIENTO:**

Las actividades de mantenimiento se agrupan en tres categorías funcionales:

- **Comprensión del software y de los cambios a realizar (Comprender):** es necesario el conocimiento a fondo de la funcionalidad, objetivos, estructura interna y requisitos del software. Alrededor del 50% de tiempo de

mantenimiento se dedica a esta actividad, a consecuencia de lo cual, las herramientas CASE incorporan utilidades que automatizan este tipo de tareas aumentando de manera notable la productividad.

- **Modificación del software (Corregir):** crear y modificar las estructuras de datos, la lógica de procesos, las interfaces y la documentación. Los programadores deben evitar los efectos laterales provocados por sus cambios. Esta actividad representa $\frac{1}{4}$ del tiempo total de mantenimiento.
- **Realización de pruebas (Comprobar):** realizar pruebas selectivas que nos aseguren la corrección del software. Vea figura 4.4.1

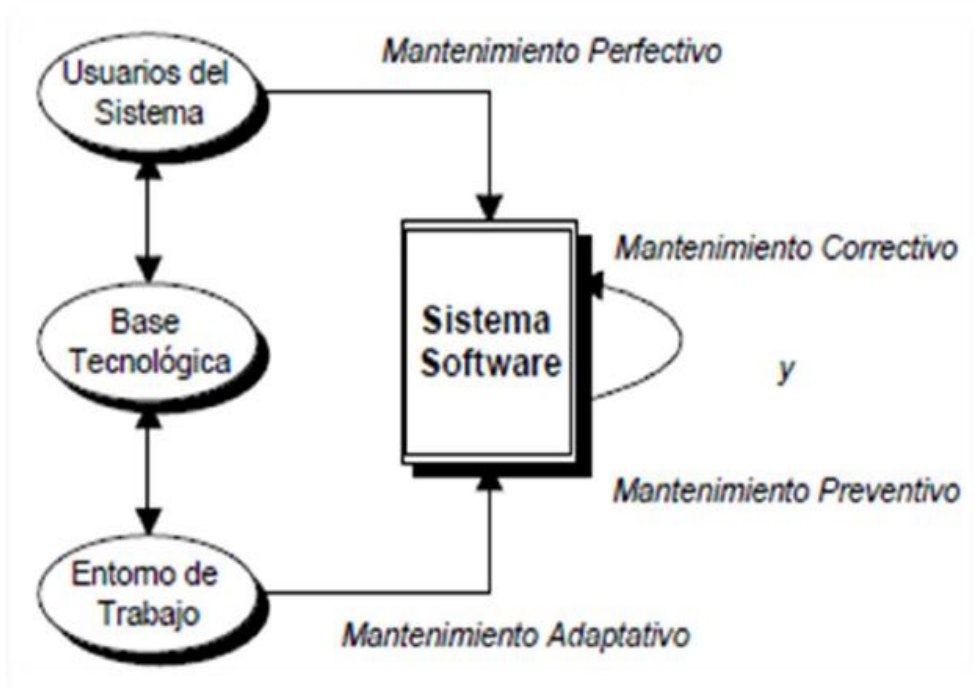


Figura 4.4.1 Representación de los tipos de Mantenimiento en Software

4.5 Generación de reportes

Los reportes se emiten por profesor-asignatura-grupo, si un profesor fue evaluado en una misma asignatura y en dos o más grupos diferentes, para ese profesor debe emitirse una evaluación por cada grupo-asignatura.

Reporte individual por factor o por escalas de evaluación

Este reporte presenta los resultados del profesor por escala, es decir la evaluación que obtiene en cada una de las agrupaciones de reactivos que forman la escala. Se indica la media de calificación que obtuvo en cada escala del cuestionario y su desviación estándar, así como el número de alumnos que lo evaluaron y la descripción de las actividades que se evalúan en cada escala. También debe de contener los valores de la media de calificación:

Casi nunca	(0.00-0.99)
Algunas veces	(1.00-1.99)
Frecuentemente	(2.00-2.99)
Casi siempre	(3.00-3.99)
Siempre	(4.00-5.00)

Donde:

- N: es igual al total de alumnos válidos que respondieron la encuesta.
- Media: Es la media de los promedios de todos los alumnos que respondieron la encuesta correctamente.
- Desviación Estándar: Es la calificación más baja que pusieron los alumnos en algún reactivo.
- Máximo: Es la calificación mas alta que pusieron los alumnos en algún reactivo.

EVALUACIÓN DE LA DOCENCIA
SEMESTRE

Vea figura 4.5.1.

PROFESOR:
CLAVE DEL
PROFESOR:
MATERIA:
CLAVE DE MATERIA:
GRUPO:

ESCALAS	N	MEDIA	DESV. EST.	MINIMO	MAXIMO
1. FORMACIÓN ÉTICA Y PROFESIONAL	28	4.98	0.06	3	5
2. ESTRATEGIAS DE ENSEÑANZA	28	4.91	0.22	1	5
3. USO DE TÉCNICAS Y RECURSOS DIDÁCTICOS	28	4.7	0.47	1	5
N = 28					

FORMACIÓN ÉTICA Y PROFESIONAL

Evalúa la frecuencia con la que el profesor enseña a respetar las opiniones de los compañeros de clase, trata a los alumnos de manera equitativa, los respeta, es ético, permite la participación de los alumnos y el dialogo abierto, evalúa el aprendizaje de manera objetiva, enseña los valores éticos de la profesión y domina los temas.

ESTRATEGIAS DE ENSEÑANZA

Valora con qué frecuencia el profesor asiste puntualmente a impartir clase, exhorta a los alumnos a respetar el horario de clase, prepara su clase y la imparte de manera clara, vincula la teoría con la práctica, realiza evaluaciones sólo de los temas revisados en clase, revisa los trabajos que deja, resuelve dudas y hace recomendaciones para que los alumnos tengan un buen desempeño en la carrera y en la vida profesional.

USO DE TÉCNICAS Y RECURSOS DIDÁCTICOS

Valora qué tan frecuentemente el profesor utiliza material didáctico o tecnológico para la enseñanza propone actividades extra clase para reforzar el aprendizaje, imparte la clase de forma dinámica, emplea diversas técnicas pedagógicas para ello y propone temas actuales para discutir en clase.

VALORES DE LA MEDIA

CASI NUNCA (0.00 - 0.99)
ALGUNAS VECES (1.00 - 1.99)
FRECUENTEMENTE (2.00 - 2.99)
CASI SIEMPRE (3.00 - 3.99)
SIEMPRE (4.00 - 5.00)

Fig. 4.5.1 Reporte por Factores.

Reporte individual por reactivo

En este reporte se muestran cada una de las actividades descritas en el cuestionario, así como la frecuencia con la que el profesor las realizó, según la opinión de los alumnos. Para interpretar los datos se debe de mostrar lo siguiente. Vea figura 4.5.2, 4.5.3, 4.5.4, 4.5.5 y 4.5.6.

Casi nunca	Algunas veces	Frecuentemente	Casi Siempre	Siempre
1	2	3	4	5
0-19%	20-39%	40-59%	60-79%	80-100%

Fig. 4.5.2 Tabla con los porcentajes según la opción seleccionada en cada pregunta.

EVALUACIÓN DE LA DOCENCIA
SEMESTRE

PROFESOR:
CLAVE DEL
PROFESOR:
MATERIA:
CLAVE DE MATERIA:
GRUPO:

PREGUNTA	CASI NUNCA	ALGUNAS VECES	FRECUENTE- MENTE	CASI SIEMPRE	SIEMPRE
	0-19%	20-39%	40-59%	60-79%	80-100%
1. Respeta a los alumnos.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
2. Domina los temas.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
3. Permite el diálogo abierto con los alumnos.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
4. Imparte su clase de forma dinámica.	0 (0%)	0 (0%)	0 (0%)	1 (33%)	2 (67%)
5. Asiste puntualmente a impartir clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
6. Vincula la teoría con la práctica.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
7. Imparte su clase de manera clara.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
8. Evalúa el aprendizaje de manera objetiva.	0 (0%)	0 (0%)	0 (0%)	1 (33%)	2 (67%)
9. Utiliza material didáctico o tecnológico para la enseñanza (lap top, cañón, diapositivas).	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
10. Resuelve las dudas de los alumnos.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
11. Enseña los valores de la profesión (ética y responsabilidad).	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
12. Emplea diversas técnicas pedagógicas para impartir la clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
13. Prepara su clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
14. Permite la participación de los alumnos.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
15. Hace recomendaciones para que los alumnos tengan un buen desempeño en la carrera y en la vida profesional.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
16. Revisa los trabajos que deja.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
17. Trata a los alumnos de manera equitativa.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
18. Enseña a respetar las opiniones de los compañeros de clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
19. Es ético (lleva a cabo lo que enseña).	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
20. Propone temas actuales para discutir en clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
21. Propone actividades extra clase para reforzar el aprendizaje.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
22. Exhorta a los alumnos a respetar el horario de clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
23. Enseña a los alumnos a comprender los temas	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
24. Realiza evaluaciones sólo de los temas revisados en clase.	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (100%)
*N = 3					

* N es igual al total de alumnos válidos que respondieron la encuesta.

Fig 4.5.3 Reporte por Reactivos.

Se muestra el promedio por reactivo según las respuestas de los alumnos.

Escalas de Evaluación

Factor I. Formación ética y profesional.

Reactivo Número	Contenido	Carga Factorial
18	Enseña a respetar las opiniones de los compañeros de clase.	.826
1	Respeto a los alumnos.	.805
17	Trata a los alumnos de manera equitativa	.793
19	Es ético (lleva a cabo lo que enseña).	.755
14	Permite la participación de los alumnos.	.700
3	Permite el diálogo abierto con los alumnos.	.695
11	Enseña los valores de la profesión (ética y responsabilidad).	.604
8	Evalúa el aprendizaje de manera objetiva.	.533
2	Domina los temas.	.559
Confiabilidad alfa de Cronbach: .94		
Varianza Explicada: 56.1%		
N=69,420		

Fig.4.5.4. Escalas para el Factor I

Factor II. Estrategias de enseñanza.

Reactivo Número	Contenido	Carga Factorial
5	Asiste puntualmente a impartir clase.	.721
22	Exhorta a los alumnos a respetar el horario de clase.	.712
13	Prepara su clase.	.656
7	Imparte su clase de manera clara.	.605

23	Enseña al alumno a comprender los temas.	.589
24	Realiza evaluaciones sólo de los temas revisados en clase.	.580
6	Vincula la teoría con la práctica	.550
16	Revisa los trabajos que deja.	.524
10	Resuelve las dudas de los alumnos.	.492
15	Hace recomendaciones para que los alumnos tengan un buen desempeño en la carrera y en la vida profesional.	.457
Confiabilidad alfa de Cronbach: .90		
Varianza explicada: 6.2%		
N = 69,420		

Fig.4.5.5. Escalas para el Factor II

Factor III. Uso de técnicas y recursos didácticos.

Reactivo Número	Contenido	Carga Factorial
9	Utiliza material didáctico o tecnológico para la enseñanza (laptop, cañón, diapositiva).	.810
21	Propone actividades extra clase para reforzar el aprendizaje.	.703
12	Emplea diversas técnicas pedagógicas para impartir la clase.	.679
4	Imparte su clase de forma dinámica	.559
20	Propone tema actuales para discutir en clase.	.486
Confiabilidad alfa de Cronbach: .82		
Varianza explicada: 3.8%		
N = 69,420		

Fig.4.5.6. Escalas para el Factor III

CONCLUSIONES

Con la elaboración del presente trabajo de tesis, podemos concluir que se realizó de manera exitosa el análisis, diseño, prueba e implementación del sistema para la evaluación docente en instituciones educativas semestralmente. Con lo cual demostramos que los conocimientos adquiridos durante la carrera de Ingeniería en Computación nos dan las bases y conocimientos necesarios para el desarrollo de cualquier sistema de cómputo.

Actualmente, el empleo de un sistema automatizado de software nos ayuda a satisfacer las necesidades del cliente, optimizando recursos, logrando la reducción de costos, mejorando el flujo de información y haciendo mucho más eficiente los procesos, en este caso la evaluación docente.

El uso de software libre permite a los desarrolladores crear una solución a bajo costo y de calidad, siendo una alternativa viable ya que éstas ofrecen rapidez de desarrollo y la opción de que otros desarrolladores aporten ideas y así vayan evolucionando conforme a las necesidades actuales.

En el desarrollo del sistema se emplearon conocimientos en áreas como redes, sistemas operativos, bases de datos y programación, todos estos adquiridos a lo largo de la carrera.

Con el desarrollo de este sistema podemos concluir que se logró:

Obtención más rápida de los resultados de las encuestas de la evaluación docente, con la cual podemos retroalimentar a los profesores en un lapso menor de tiempo, con resultados más precisos, que evalúen las características y habilidades del profesor.

La administración del sistema es simple y confiable.

Mejóro la reducción de costos tanto en materiales como personal administrativo

Se pueden obtener reportes específicos de cada profesor para realizar un análisis estadístico identificando qué habilidades deben de mejorarse.

Aumentar la participación de los alumnos, ya que pueden realizar la evaluación desde cualquier lugar con acceso a internet, sin estar limitados a un horario, pudiendo realizarla en una o varias sesiones, respetando el periodo de la evaluación.

La interfaz es de fácil manejo por lo que los alumnos no tienen problemas al realizar la encuesta.

Realizarlo en equipo amplió la búsqueda de soluciones, ya que todos los miembros del equipo aportan ideas que ayudan a encontrar la solución más óptima y viable.

Los objetivos de la tesis se cumplieron de manera exitosa ya que se pudo conocer las características del desempeño de los profesores en las diferentes asignaturas impartidas en la institución, así como identificar sus fortalezas y áreas en las que, según la opinión de los alumnos, deban mejorar.

BIBLIOGRAFIA

- Alarcón, Raúl, "Diseño Orientado a Objetos con UML", Madrid, 2000
- Howard, Podeswa, "Programación UML", Madrid, España, Ed. Anaya Multimedia, 2010.
- Silberschatz A., Korth H. y Sudarshan S, "Fundamentos de Bases de Datos", Madrid, 2002.
- Vinculación de los fundamentos filosóficos del método de simulación con la Modelación y estrategia: algunas consideraciones desde una perspectiva pedagógica (2002).
- Regla Alicia Sierra Salcedo, Pág. 319.
- Metodología de la Investigación educacional. Primera Parte (1996). Gastón Pérez Rodríguez, Gilberto García Batista, Irma Nocedo de León y Miriam Lucy García Inza. Pág. 41.
- Qué piensan los profesores de cómo se debe enseñar. José Fernández González. I.B. "Cabrera Pinto", La Laguna, Escuela Universitaria de Formación del Profesorado, Universidad de La Laguna. Nicolás Elortegui Escartín. I.P.F.P. De Santa Cruz de Tenerife.
- García Aretio, Lorenzo. (2001) La Educación a Distancia. "De la teoría la práctica". Edith. Ariel. Madrid
- Florez Ochoa, Rafael. Hacia Una Pedagogía Del Conocimiento, McGraw Hill, 1994, Santa Fé De Bogota. Pág. 154,160, 161
2. Ibid Pág. 167-174, los modelos pedagógicos son retomados de este texto
- El lenguaje de unificado de modelado. de Grady Booch, James Rumbauch e Ivar Jacobson. Edit:Addison Wesley. 1º edición en español 1999
- Modelos y diseños orientados a objetos - Metodología OMT - . De James Rumbaugh, Michael Blaha, William Premeralni, Frederick Hedí y William Lorensen. Editorial: PRENTICE HALL. 1996.
- Curso de Ingeniería del Software - Unidad 5.1 Diseño de Bases de Datos Relacionales y Conceptos de la Orientación a Objetos - . De J. Pedro Caraca, Valiente y Hernández. ITBA (Instituto Tecnológico de Buenos Aires - Universidad Privada -). 1997.
- Grupo Isaias Carrillos Perez. (2008) Metodología del desarrollo de software. New York, Editorial Edit and write.
IBM, Rational Software. (2003). Rational Rapid Developer, Technical Overview, EE.UU: IBM publications, World Wide Web.
- ITSA (2008). Metodologías De Desarrollo De Software, Canada, Editorial Canada Pen.

- Jacobson, I., Booch, G., Rumbaugh J. (2000) Proceso Unificado de Desarrollo de Software, New York:Editorial Mc Graw Hill.
- Kruchten, P. (1995). Architectural Blueprints—the “4+1” View Model of Software Architecture. IEEE Software.
- Marcos, E. (2005). “Investigación en Ingeniería del Software vs. Desarrollo Software”, Grupo KYBELE, Universidad Rey Juan Carlos.
 - Morgan, Stanley (2001). Sr. Business Analyst. New York: Editorial Prentice Hall.
 - Pressman, Roger. (2003) Ingeniería de Software 6ta edic. New York Editorial Mc Graw Hill.
 - RUP/Easy. (2004). GUÍA METODOLÓGICA DE DESARROLLO de SISTEMAS
 - Schmuller, Joseph. (2000) Aprendiendo UML en 24 horas. México: Editorial Prentice Hall.
 - Metodologías RUP y XP - [PROCESOS DE DESARROLLO] Autor: Jack (Blogger)
 - Barrientos Enríquez, Aleida Mirian. El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML.
 - Jacobson, Ivar; Booch, Grady; Rumbaugh, James. El Proceso Unificado de Desarrollo de Software.
 - Echevarría Cossío, Yanelis. Modelo Ágil de Desarrollo de Proyectos de Software:Paradigma 3P.
 - Introducción a los sistemas de bases de datos, Prentice Hall, 2001 [7ª edición]. Ramez A. Elmasri & Shamkant B. Navathe:
 - Fundamentos de Sistemas de Bases de Datos, Addison-Wesley, 2007 [5ª edición].Thomas M. Connolly & Carolyn E. Begg:
 - Sistemas de Bases de Datos; Addison-Wesley, 2005 [4ª edición]. Henry F. Korth, Abraham Silberschatz & S. Sudarshan:
 - Fundamentos de Bases de Datos, McGraw-Hill, 2006 [5ª edición].Olga Pons, Nicolás Marín, Juan Miguel Medina, Silvia Acid
 - Introducción a los sistemas de bases de datos, Prentice Hall, 2001 [7ª edición]. Ramez A. Elmasri & Shamkant B. Navathe:
 - Fundamentos de Sistemas de Bases de Datos, Addison-Wesley, 2007 [5ª edición].Thomas M. Connolly & Carolyn E. Begg:
 - Sistemas de Bases de Datos; Addison-Wesley, 2005 [4ª edición]. Henry F. Korth, Abraham Silberschatz & S. Sudarshan:
 - Fundamentos de Bases de Datos, McGraw-Hill, 2006 [5ª edición].Olga Pons, Nicolás Marín, Juan Miguel Medina, Silvia Acid &
 - Castillo Arredondo, S. (2002). Compromisos de la Evaluación Educativa. Editado por Practice Hall, Madrid - España.
 - Hernández Sampieri, R., Fernández Collado, C. y Baptista Lucio, P. (2006).

- Metodología de la investigación. Cuarta edición, editado por McGraw – Hill Interamericana Editores, S.A. DF – México.
- Imídeo G. Nérici (1985). Metodología de la Enseñanza. Colección Actualización Pedagógica. Editorial Kapelusz – Mexicana. México.
Jorba, J. y Sanmartí, N. (2000). La función pedagógica de la evaluación. En: evaluación como ayuda al aprendizaje. Margarita Ballester (et al) Editorial Grau, Barcelona, 2000. pp. 23-24.
- Mateo Andrés, J. (2000). La evaluación educativa, su práctica y otras metáforas. Editado por la Editorial Horsori, Universidad de Barcelona-España.

LINKOGRAFÍA

<http://www.ub.es/geocrit/b3w-207.htm>

<http://www.quadernsdigitals.net/>

<http://www.uib.es/depart/gte/revelec.html>

[http://184.182.233.153/rid=1JDJ1SFVP-](http://184.182.233.153/rid=1JDJ1SFVP-X2JLXJ11MG/Evaluacion_de_la_calidad_docente.pdf)

[X2JLXJ11MG/Evaluacion_de_la_calidad_docente.pdf](http://184.182.233.153/rid=1JDJ1SFVP-X2JLXJ11MG/Evaluacion_de_la_calidad_docente.pdf)

<http://www.doredin.mec.es/documentos/007200330499.pdf>

<http://dialnet.unirioja.es/servlet/articulo?codigo=1400895>

http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1607-40412008000300006

http://www.dgb.sep.gob.mx/informacion_academica/pdf/evaluacion-docente-documento2010.pdf

<http://www.itera.com.mx/itera/productos/fundamentos.asp>.

<http://jackopc.blogspot.com/2007/05/metodologias-rup-y-xp-procesos-e.html>

<http://lucas.hispalinux.es/Tutoriales/doc-modelado-sistemas-UML/multiple-html/index.html>

<http://www.cs.ualberta.ca/~pfiguero/soo/metod/uml-met.html>

<http://www.mcc.unam.mx/~cursos/Objetos/Cap8/cap8.html>

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

http://www.sparxsystems.com.au/UML_Tutorial.htm

http://www.togethersoft.com/services/practical_guides/umlonlinecourse/index.html

<http://www.my-project-management-expert.com/the-advantages-and-disadvantages-of-rup-software-development.html>