

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA**



SISTEMA INTEGRAL PARA LA EVALUACIÓN DE
ALUMNOS A DISTANCIA

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

**FELIPE DE JESÚS AVILA VALENCIA
CAROLINA HUITRÓN VÁZQUEZ**



DIRECTOR DE TESIS:
ING. JESÚS JAVIER CORTES ROSAS



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.

Introducción	3
Capítulo 1	4
La evaluación en línea	4
1.1 Antecedentes	4
1.2 La evaluación	5
1.3 Tipos de evaluación	7
1.4 Procesos o momentos de la evaluación	8
1.5 Evaluación a distancia	9
1.5.1 Ventajas y desventajas de la evaluación a distancia	11
Capítulo 2	12
Conceptos básicos	12
2.1 Ingeniería del software	12
2.1.1 El ciclo de vida clásico	13
2.1.2 Modelos de ciclos de vida más comunes	21
2.2 Arquitectura del software	26
2.3 DFD (Diagrama de Flujo de Datos)	29
2.3.1 Definición de DFD	29
2.3.2 Características Principales de DFD	33
2.3.3 ¿Por qué Utilizar DFD?	34
2.4 Sitios Web	35
2.4.1 ¿Qué es un servidor Web?	36
2.4.2 Protocolo HTTP	37
2.4.3 Lenguaje HTML	38
2.4.4 Lenguaje de programación PHP	40
2.4.5 Características de JavaScript	42
2.4.6 Manejadores de bases de datos	44
2.4.7 MySQL	48
Capítulo 3	52
Identificación y contexto del sistema	52
3.1 Antecedentes	52
3.2 Planteamiento del problema	53
3.3 Objetivos generales	54
3.4 Objetivos particulares	54
3.5 Alcances y Limitaciones del Sistema	55
3.6 Descripción de las alternativas de solución	56
Capítulo 4	60
Análisis del sistema	60
4.1 Análisis de los requerimientos del sistema	60
4.1.1 ¿Qué es lo que el usuario desea?	60

4.1.2 Clasificación de los requerimientos	61
4.2 Identificación de los tipos de usuarios que harán uso del sistema	65
4.3 Estudio de los riesgos	65
4.4 Etapas de desarrollo del sistema	67
4.4.1 Módulo de diseño	68
4.4.2 Módulo de aplicación	72
4.4.3 Módulo de administración	74
Capítulo 5	80
Diseño del sistema	80
5.1 Diseño e implementación de diagramas de flujo de datos	80
5.1.1 Diagrama de flujo de datos	80
5.1.2 Diagramas de flujos de datos aplicados al sistema	81
5.2 Creación de base de datos	88
5.2.1 Tablas derivadas	88
5.2.2 Diseño del Modelo Entidad-Relación	91
5.3 Diseño de la arquitectura de los módulos del sistema	93
Capítulo 6	95
Programación y pruebas del sistema	95
6.1 Estudio del software utilizado para el desarrollo: EasyPHP	95
6.1.1 ¿Qué es EasyPHP?	95
6.2 Manual de usuario del sistema	97
6.2.1 Tipos de sesión	97
6.2.2 Módulo de administración	103
6.2.3 Módulo de diseño	120
6.2.4 Módulo de aplicación	137
6.3 Pruebas del sistema	143
Conclusiones	150
Diccionario de datos	151
Índice de gráficas	164
Bibliografía	166

Introducción

Este trabajo escrito surge como una propuesta de solución a una necesidad específica: crear una herramienta que sirva de apoyo en el proceso de evaluación. Esta herramienta se tradujo en el desarrollo de un sistema informático que permitirá lograr el objetivo deseado.

Asimismo, es importante mencionar que la institución que hizo expresa la necesidad arriba mencionada fue la Escuela Nacional Preparatoria (ENP), plantel número 7 “Ezequiel A. Chávez”, perteneciente a la Universidad Nacional Autónoma de México (UNAM), a través del Ing. Alejandro Villagómez, quien funge como profesor en el plantel citado, y que además, es el coordinador del área de cómputo del plantel. También el Ing. Alejandro fue el encargado de proporcionar todos los requerimientos e información relacionada con el desarrollo de este proyecto.

En el transcurso de este documento se enlistan y describen claramente todos los requerimientos que se ven involucrados con la meta citada. Sin embargo, se puede adelantar que se busca agilizar el proceso de diseño, aplicación y obtención de resultados de los exámenes que se realizan en la institución que hizo expresa la necesidad en cuestión. Como se ha mencionado antes, la solución está basada con el desarrollo de un sistema de cómputo, el cual se ha solicitado tenga un enfoque práctico, un orden lógico y de fácil manejo que permita la rápida aceptación de todos los usuarios involucrados.

Si bien es cierto que se trata de un problema que no es nuevo, también lo es el hecho de que hablamos de necesidades particulares que requieren de una solución hecha a la medida para lograr cubrir todos los requisitos que son la razón de este trabajo.

Capítulo 1

La evaluación en línea

1.1 Antecedentes

La evaluación hoy en día es quizá uno de los temas con mayor protagonismo en el ámbito educativo, y no porque se trate de un tema nuevo en absoluto, sino porque profesionistas, educadores, padres, alumnos y toda la sociedad en su conjunto, son más concientes que nunca de la importancia y las repercusiones del hecho de evaluar o de ser evaluado. Existe quizá una mayor conciencia de la necesidad de alcanzar determinadas cotas de calidad educativa, de aprovechar adecuadamente los recursos, el tiempo y los esfuerzos y, por otra parte, el nivel de competencia entre los individuos y las instituciones también es mayor.

En palabras de A. de la Orden (1989): "la evaluación, al prescribir realmente los objetivos de la educación, determina, en gran medida [...] lo que los alumnos aprenden y cómo lo aprenden, lo que los profesores enseñan y cómo lo enseñan, los contenidos y los métodos; en otras palabras, el producto y el proceso de la educación [...] querámoslo o no, de forma consciente o inconsciente, la actividad educativa de alumnos y profesores está en algún grado canalizada por la evaluación".

Los factores antes mencionados, han llevado a una *cultura de la evaluación* que no se limita a la escuela sino que se extiende al resto de las actividades sociales. La ampliación del ámbito de la evaluación desde los resultados y procesos del aprendizaje de los alumnos, la práctica docente, el sistema educativo en su conjunto y de cualquier persona en el mundo laboral ha

dibujado en los últimos años un nuevo escenario para las prácticas evaluativas, que se han desarrollado a todos los niveles de manera muy importante.

Como ya se ha explicado y como consecuencia de la multiplicación de instituciones de educación en diferentes niveles y en varias partes del mundo, entre ellas México, ha surgido la preocupación sobre la calidad de los egresados de tales instituciones. Con el paso del tiempo se han llegado a concretar proyectos de creación de organismos evaluadores con finalidades muy particulares; un ejemplo de esto en nuestro país, es el Centro Nacional de Evaluación para la Educación Superior (Ceneval). El cual *“es una asociación civil sin fines de lucro cuya actividad principal es el diseño y aplicación de instrumentos de evaluación de conocimientos, habilidades y competencias, así como el análisis y la difusión de los resultados que arrojan las pruebas”* [CENEVAL, 2007].

En décadas pasadas se tuvieron diferentes formas de resolver los problemas y de evaluarlos de forma directa. Actualmente, el desarrollo social, científico y tecnológico ha producido nuevas situaciones en las que los esquemas tradicionales de evaluación resultan insuficientes. Un caso específico, y motivo de este trabajo escrito, es la evaluación a distancia. Sin embargo, antes de hablar sobre el tema, procederemos a explicar el concepto de evaluación.

1.2 La evaluación

Es importante, previo a dar una definición de lo que es la evaluación, mencionar que existe una diferencia muy importante entre los conceptos de evaluación y calificación, es decir, éstos no son sinónimos uno del otro aunque en muchas ocasiones se utilizan de manera indistinta.

La evaluación es un proceso que va más allá de una simple medición. Implica descripciones cuantitativas y cualitativas de la conducta del alumno, la interpretación de dichas descripciones y por último la formulación de juicios de valor basados en la interpretación de las descripciones.

El término calificación está referido exclusivamente a la valoración de la conducta de los alumnos. Calificar, por tanto, es una actividad más restringida que evaluar. La calificación será la expresión cualitativa (apto/no apto) o cuantitativa (10, 9, 8, etc.) del juicio de valor que emitimos sobre la actividad y logros del alumno. En este juicio de valor se suele querer expresar el grado de suficiencia o insuficiencia, conocimientos, destrezas y habilidades del alumno, como resultado de algún tipo de prueba, actividad, examen o proceso.

Se evalúa siempre para tomar decisiones. No basta con recoger información sobre los resultados del proceso educativo y emitir únicamente un tipo de calificación, si no se toma alguna decisión, no existe una auténtica evaluación. Así pues, la evaluación es una actividad o proceso sistemático de identificación, recogida o tratamiento de datos sobre elementos o hechos educativos, con el objetivo de valorarlos primero y, sobre dicha valoración, tomar decisiones.

Valorar significa otorgar una apreciación justa a una cualidad, una actitud, una persona, una cosa, etc. tomando como base las manifestaciones externas (por lo tanto objetivas) que nos sirven como señales para poder elaborar nuestro juicio. Formular un juicio de valor significará, por tanto, emitir una apreciación justa acerca de la conducta (del alumno) que estamos analizando en la evaluación.

“Toda evaluación supone una medición previa acerca de aquellos rasgos que son importantes para juzgar la conducta que pretende evaluarse. La medición

se realizará a través de pruebas, escalas, entrevistas, etc. que describan los rasgos seleccionados en forma efectiva y precisa. Pero hasta que el resultado de dicha medición es interpretado y valorado puede decirse que se ha completado el proceso de la evaluación” [Red Escolar SEP-ILCE, 2007].

1.3 Tipos de evaluación

Para definir los tipos de evaluación existentes hay que analizar la evaluación desde una perspectiva cuantitativa que tiene que ver con el concepto de evaluación como medición y desde una perspectiva cualitativa, que relaciona la evaluación con la capacidad y la acción de apreciar, valorar, comparar y comprender. Existen dos paradigmas sobre evaluación:

- Paradigma positivista (cuantitativo): totalmente objetivo, centrado en resultados numéricos o porcentuales. Utiliza instrumentos objetivos, encasilla los resultados en términos medibles y cuantificables, por ello se instrumenta de las matemáticas y la estadística. Sus resultados son generales, no permite ver las diferencias individuales en cada estudiante, es decir lo que interesa aquí es producir información que sea de utilidad para el control.

- Paradigma alternativo o naturalista (cualitativo): adopta una orientación cualitativa y surge de las críticas y limitaciones del paradigma positivista. Desde esta perspectiva se valorizan los aspectos contextuales y situacionales, de ahí que lo que caracteriza a este paradigma es su preocupación por indagar en profundidad, atendiendo a la complejidad de variables, su carácter heurístico y la posición del estudiante que va a actuar de acuerdo con sus propios conocimientos y creencias por lo cual no se puede exigir objetividad y neutralidad absoluta. Se considera que

desde la evaluación es posible poner en marcha procesos de reflexión crítica; procesos de toma de decisiones libres y procesos de elaboración de intenciones y proyectos en un marco de relaciones interpersonales. No utiliza métodos estadísticos, sino fenomenológicos, etnográficos, históricos, estudios de caso.

Según el paradigma que guíe al docente se citan dos tipos de evaluación:

- Evaluación Cuantitativa: cuyos resultados son medibles, expresados en números y porcentajes. Utiliza pruebas objetivas, de ensayo u otros instrumentos objetivos que permiten ver el rendimiento en general.
- Evaluación Cualitativa: que evidencia mejor los logros en el proceso del aprendizaje en cada estudiante. Sus instrumentos son preguntas orientadoras que a través de estudios de caso, entrevistas, grupos focales permiten establecer categorías, vivencias o dominios que intervienen durante la formación del estudiante.

Cabe resaltar que a la fecha no se puede decir que un tipo de evaluación sea mejor que otro, su uso dependerá de la naturaleza y propósito del curso a desarrollar. Sin embargo es muy productivo utilizar ambos tipos de evaluación durante el proceso de aprendizaje.

1.4 Procesos o momentos de la evaluación

En todo proceso de aprendizaje el docente debe considerar los siguientes momentos de la evaluación:

- Evaluación inicial o diagnóstica: se realiza al inicio del curso o de una sesión educativa con el fin de conocer los conocimientos previos que trae el alumno y así orientar la profundidad y forma de desarrollar los contenidos.
- Evaluación de proceso o formativa: permite hacer la retroalimentación respectiva, porque durante el proceso de aprendizaje se van identificando las limitaciones, avances y reajustes en el desarrollo del curso.
- Evaluación final o sumativa: permite ver el logro de objetivos o competencias y determina la nota (cuantitativa o cualitativa) de aprobación o no del curso.

1.5 Evaluación a distancia

Tomando en consideración todos los elementos mencionados hasta este momento, podemos decir que la educación y/o evaluación a distancia es la respuesta al hecho social que ha tomado un papel relevante en la actualidad: la continua preocupación por la necesidad y derecho de una educación permanente.

La educación a distancia es una alternativa válida, ya que facilita estrategias de educación permanente. Normalmente la población que atiende son adultos que quieren iniciar o continuar estudios, o graduados que buscan su renovación o mejoramiento, sin sacarlos de su contexto laboral, social y familiar. Esto último, permite superar la clásica barrera entre la escuela y la vida, puesto que el estudiante no es separado de su medio. Además, dada la amplia cobertura

social que puede alcanzar, hace más real la igualdad de oportunidades, por lo que se transforma en una respuesta a la demanda democrática de educación.

La educación y la evaluación a distancia, no es un concepto nuevo. Hasta este momento ya se han realizado varias implementaciones de este modelo en diferentes niveles educativos. Sin embargo, aun es vista con desconfianza por la mayor parte de la población.

Definamos entonces qué es la educación a distancia, específicamente la enseñanza en línea, siendo ésta última la que es motivo de nuestro trabajo: “es una forma de educación a distancia, un proceso que incluye cursos dictados a través de correo electrónico, video, o conexión vía teléfono o TV por satélite – cualquier forma de aprendizaje que no implique la clase tradicional en la cual los estudiantes y el instructor deben estar en el mismo lugar al mismo tiempo” [KOROS, 2001].

Con base en lo anterior, debe entenderse que existe evaluación a distancia siempre que la comunicación profesor-alumno se realice a través de otros medios que no sea la expresión corporal o la palabra oral sin estar ambas partes físicamente en el mismo lugar.

El uso de las tecnologías de la información y la comunicación aplicadas a la educación a distancia ha generado un proceso de reflexión sobre cómo evaluar los aprendizajes, permitiendo encuentros desde diferentes lugares, al mismo tiempo, o en diferente tiempo. Cuando han estado presentes han contribuido al desarrollo y mejora de los procesos de enseñanza-aprendizaje. Sin embargo es importante señalar que ellas requieren que tanto los profesores como los estudiantes desarrollen habilidades y competencias que permitan aprovechar estas herramientas y mejorar su efectividad.

1.5.1 Ventajas y desventajas de la evaluación a distancia

Ventajas

- Separación física del alumno-profesor, lo que reduce posibles problemas de espacio y tiempo.
- Economía de recursos, se reducen los recursos materiales necesarios tanto para instituciones, profesores y alumnos.
- Mayor rapidez en la actualización de los contenidos a evaluar.
- No existe la improvisación. Los contenidos a evaluar deben ser preparados con anticipación.
- La evaluación es realizada en tiempo real por lo que el alumno sabrá en ese instante sus resultados.

Desventajas

- Poca o nula supervisión del profesor.
- No es posible una evaluación personalizada.
- El tiempo de respuesta para contestar alguna duda puede llegar a ser considerable.

Capítulo 2

Conceptos básicos

2.1 Ingeniería del software

Es importante tener en cuenta que para el desarrollo de todo sistema se necesita establecer un plan de trabajo que permita un enfoque sistemático del software a desarrollar para un mejor análisis, diseño, y codificación del mismo, para ello se requiere tener los conocimientos específicos y concisos en el área de computación que serán la base para la implementación del sistema.

En todo proyecto existe una fuerte comunicación entre el cliente y el desarrollador, como se ha mencionado antes, “El sistema integral para la evaluación de alumnos a distancia” es un proyecto solicitado por la Escuela Nacional Preparatoria “Ezequiel A. Chávez” Plantel 7 (ENP), por lo que las especificaciones y tareas del sistema estarán determinadas por el coordinador de informática en dicho plantel, con la intención de definir los recursos, el tiempo y requerimientos del sistema de acuerdo a las necesidades académicas que se presenten en la institución, haciendo uso de la ingeniería de software para la realización de dicho sistema.

Antes de dar una definición concisa de lo que es la ingeniería del software es necesario tener presente el concepto único de software, por lo que a continuación se cita la siguiente definición citada por Lewis "software es la suma total de los programas de computadora, procedimientos, reglas, la documentación asociada y los datos que pertenecen a un sistema de cómputo".

Por consiguiente es posible considerar que la ingeniería de software es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr elaborar soluciones correctas, utilizables y de bajo costo.

“El proceso de ingeniería de software se lleva a cabo por etapas parcialmente ordenadas en donde las necesidades del usuario son analizadas mediante una serie de preguntas y respuestas traducidas en requerimientos de software, estos requerimientos son transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso. Durante el desarrollo de dicho proceso se define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un determinado objetivo, dicho proceso requiere de un conjunto de conceptos, una metodología y un lenguaje propio” [BOOCH, 1996].

A este proceso también se le llama el ciclo de vida del software que comprende cuatro grandes fases:

- Concepción.- Define el alcance del proyecto.
- Elaboración.- Define un plan del proyecto.
- Construcción.- Crea el producto.
- Transición.- Transfiere el producto a los usuarios.

2.1.1 El ciclo de vida clásico

El ciclo de vida del software se puede dividir en tres fases: una de planificación, otra de desarrollo y una tercera de mantenimiento, que engloba a las seis etapas (Ingeniería del sistema, Análisis de los requisitos, Diseño, Codificación, Pruebas y Mantenimiento) tradicionales del ciclo de vida.

La fase de Planificación del software comprende las etapas de Ingeniería del sistema o Análisis del sistema, estableciendo los requisitos del software y el análisis de los requerimientos, que se traduce en una especificación de requerimientos. La fase de desarrollo comprende las etapas de Diseño, Codificación y Pruebas, finalmente está la fase de mantenimiento.

2.1.1.1 Análisis del sistema

Esta etapa tiene por objeto realizar un análisis global del sistema, estableciendo los requisitos de todos los elementos, identificando las necesidades del cliente y realizando un análisis técnico y económico del mismo, haciendo un planteamiento del coste-beneficio, estableciendo restricciones de coste y tiempo, asignando funciones al software, hardware, al personal, a las bases de datos y a otros elementos del sistema.

Todo el análisis del sistema se recoge en un documento denominado "especificación del sistema", que sirve como base para la ingeniería del hardware, del software, de las bases de datos y de la ingeniería humana. Este documento describe toda la información que se ha obtenido en esta fase sobre el sistema que se quiere implementar. Un ejemplo de lo que se requiere es como a continuación se plantea:

a) Objetivos generales del sistema

- Fines del sistema.

- Funcionamiento y rendimiento requerido.
- Entradas al sistema.
- Salidas del sistema.
- Restricciones.

b) Definición de Requisitos del sistema

- Software.
- Hardware.
- Base de datos.
- Personal y otros elementos del sistema.

c) Análisis Técnico

- Rendimiento.
- Fiabilidad.
- Mantenimiento.
- Producción.

d) Análisis Económico

- Costes de producción.
- Costes de mantenimiento.
- Relación Coste/beneficio.

e) Viabilidad del sistema

- Economía y de Mercado.
- Técnica.

f) Especificación de la arquitectura del sistema

- Diagrama de contexto de la arquitectura.
- Diagrama de flujo de la arquitectura.
- Descripción de los subsistemas.

2.1.1.2 Análisis de los requisitos del software

En el análisis de los requisitos del sistema el ingeniero de software determina los flujos de información, las estructuras de datos, las características funcionales del sistema, los requerimientos, rendimiento y restricciones impuestas por el cliente. En este proceso se identifican cinco actividades que se enuncian a continuación:

Reconocimiento del problema software

Dentro de esta actividad es necesario identificar el software que se desarrollara dentro del contexto del sistema.

Evaluación del problema y síntesis de la solución

El analista o ingeniero de programación deberá establecer las características de la interfaz del sistema y describir las restricciones del diseño.

Modelado

En esta parte del proceso se pretende crear un borrador en papel que modele la función y comportamiento del sistema.

Especificación

En ella se establece una descripción completa y funcional de la información, haciendo referencia al rendimiento y diseño del sistema.

Revisión de la especificación

El ingeniero del software, el cliente, el experto y los usuarios efectúan una revisión de la especificación acerca de los requisitos del software, al finalizar

esta revisión el cliente y el ingeniero de sistema firmarán dicho acuerdo para dar inicio al desarrollo del sistema.

2.1.1.3 Diseño del sistema

Esta etapa del proceso representa la primera fase del desarrollo del sistema en donde se crea un modelo que mas adelante será construido, en ella se estudia la parte del diseño preliminar y el diseño detallado.

2.1.1.3.1 Diseño Preliminar

En el diseño preliminar se desarrolla una estructura funcional al definir las interfaces que existirán entre los módulos estableciendo las estructuras de datos. El diseño preliminar consiste en tres partes:

- Diseño de datos.
- Diseño arquitectónico.
- Diseño de la interfaz hombre-máquina.

Principios para especificar y diseñar los datos

- Identificar todas las estructuras de datos.
- Estructurar un diccionario de datos que ayude a definir el diseño de los datos y del sistema.
- El lenguaje de programación debe soportar la especificación y realización de tipos abstractos del sistema.

- Desarrollar una estructura de datos de manera que se puedan reutilizar en el desarrollo de la aplicación.

Diseño arquitectónico

Esta parte del diseño mezcla la estructura de programa y la de los datos al definir las interfaces que faciliten el flujo de datos entre los módulos del software.

Diseño de la interfaz hombre-máquina

El objetivo de esta etapa es diseñar una interfaz que comunique al usuario con el software del sistema durante su ejecución, estableciendo las características y restricciones que deberá tener dicha interfaz, así como la estructura que represente el perfil de un usuario final.

Una buena interfaz deberá tomar en cuenta los siguientes principios:

- Mostrar la información que es relevante para el usuario.
- Hacer uso de gráficos sin saturar la interfaz de solo texto.
- Enviar mensajes de error.
- Preguntar por la verificación de cualquier acción que pueda modificar o destruir la información que se esta introduciendo.
- Permitir deshacer la ejecución de las acciones.
- Proteger al sistema de los errores del usuario que puedan modificar o causar algún daño al sistema.
- El significado de una pantalla debe ser claro para el usuario.

2.1.1.3.2 Diseño detallado

Es la representación algorítmica del software o diseño procedimental, en esta etapa se realiza el diagrama de cada módulo y las especificaciones procedimentales. El diseño es la parte más importante de la ingeniería del software por lo que es necesario realizar el análisis correcto para obtener un resultado exitoso.

2.1.1.3.3 Codificación

En esta etapa se realiza la programación del modelo aceptado y aprobado por los analistas utilizando un lenguaje apropiado de programación, como resultado de este proceso se obtiene un listado de código fuente el cual debe documentarse incorporando comentarios al principio de cada módulo codificado haciendo énfasis de las características de la interfaz y de los argumentos que tenga.

La presentación de la estructura de código fuente es una ayuda importante a la legibilidad del mismo, es por ello que el ingeniero de software realiza las construcciones lógicas y los bloques de código.

2.1.1.3.4 Pruebas del software

Esta etapa representa una revisión final de las especificaciones, del diseño y de la codificación que constatan si el software desarrollado satisface las especificaciones del cliente. Durante todos los procesos que conforman al ciclo de la vida del software se realizan pruebas de unidad, integración y validación. Las pruebas se pueden realizar de dos formas:

Pruebas de caja negra

Se llevan a cabo sobre la interfaz del software y tiene como objetivo demostrar que el software funciona correctamente verificando que las entradas se aceptan de forma adecuada y que se produce una salida correcta. Este tipo de pruebas no toman en cuenta la estructura lógica interna del software.

Pruebas de caja blanca

Se comprueban los caminos lógicos del software analizando bloques específicos de código (bucles, sentencias, etc.).

Generalmente se realiza una combinación de las pruebas de caja negra con las pruebas de caja blanca con la finalidad de validar la interfaz del software y asegurar el funcionamiento del sistema. El proceso general de desarrollo del sistema y el de las pruebas se contemplan como un proceso continuo

2.1.1.3.5 Mantenimiento

El software generado en la fase de Desarrollo debe ser mantenido debido a que se puede tener la necesidad de modificarlo después de ser entregado al cliente, estas modificaciones ocurrirán debido a:

- Errores encontrados.
- Cambios en el entorno externo al que dicho software debe adaptarse.
- Modificaciones o actualizaciones que requiera el cliente.

La etapa de mantenimiento es considerada como la más importante dentro de una organización ya que puede generar gastos considerables a la empresa si se utiliza una mala ingeniería de desarrollo del software y las personas que participaron no están disponibles para mantenerlo.

2.1.2 Modelos de ciclos de vida más comunes

El ciclo de vida que se seleccione en un proyecto [Davis, 1988], influye en el éxito obtenido en el sistema debido a que con esto es posible aumentar la velocidad de desarrollo, mejorar la calidad, el control y el seguimiento del proyecto, así como la oportunidad de minimizar gastos y riesgos, o simplemente mejorar las relaciones con los clientes. Una selección inadecuada del ciclo de la vida puede ocasionar que el desarrollo del sistema sea lento, innecesario y frustrante.

Existen varios modelos que representan el ciclo de vida de un proyecto, por lo que a continuación se enuncian los más estudiados y utilizados por los ingenieros de software:

- Modelo en cascada.
- Modelo incremental.
- Modelo en espiral.

2.1.2.1 Modelo en cascada

El número de fases o etapas que se proponen en este ciclo son: análisis de requisitos del sistema, análisis de requisitos del software, diseño preliminar, diseño detallado, codificación, pruebas, explotación y mantenimiento. Es

considerado el ciclo de vida más antiguo y utilizado. Dentro de sus características más importantes de este modelo son las siguientes:

- Cada fase empieza cuando se ha terminado la fase anterior.
- Para pasar de una fase a otra es necesario conseguir todos los objetivos de la etapa previa.
- Ayuda a prevenir que se sobrepasen las fechas de entrega y los costes esperados.
- Al final de cada fase el personal técnico y los usuarios tienen la oportunidad de revisar el progreso del proyecto.

Con el paso del tiempo se han encontrado varias desventajas al utilizar el modelo de cascada, de las cuales sobresalen las siguientes:

- No refleja el proceso real de desarrollo de software.
- Se tarda mucho tiempo en pasar por todo el ciclo, recordando que se tendrá que finalizar una fase para poder pasar a la siguiente.
- El sistema no estará disponible hasta las fases finales del proyecto.

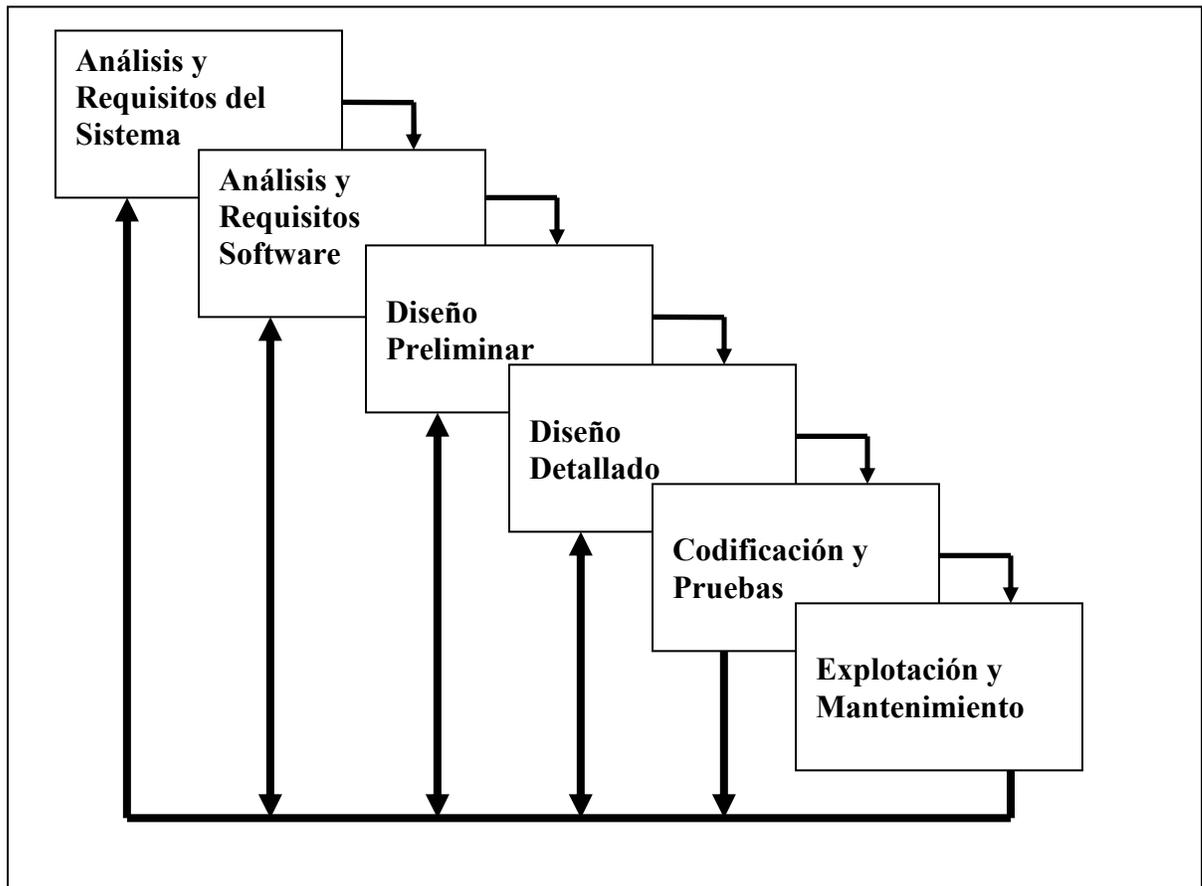


Fig. 2.1 Ciclo de vida en cascada

2.1.2.2 Modelo incremental

En este modelo se va creando el software del sistema añadiendo componentes o incrementos necesarios para un mejor funcionamiento del sistema; éste no tendrá una fecha fija de entrega debido a la integración continua de nuevos elementos que darán mayor funcionalidad al sistema. Representa un avance sobre el modelo en cascada, aunque también presenta ciertos problemas porque aunque permite el cambio de requisitos, no se sabe de manera certera si serán válidos, lo que puede generar errores que al ser detectados deberán ser corregidos lo que genera un aumento en el presupuesto del proyecto.

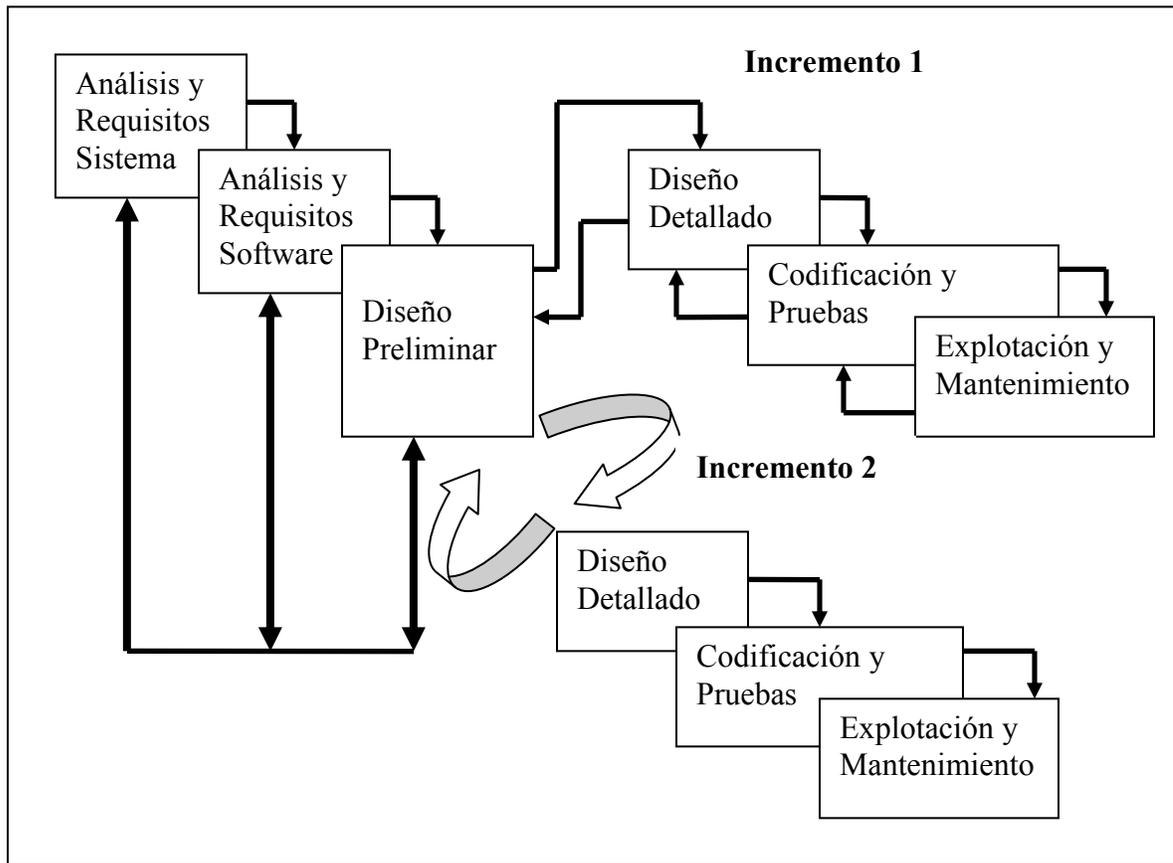


Fig. 2.2 Modelo incremental

2.1.2.3 Modelo en espiral

Este modelo consta de una serie de ciclos, cada uno inicia identificando los objetivos, las alternativas y las restricciones del mismo. Cada ciclo de la espiral comienza con la identificación de los objetivos, el siguiente paso es evaluar las diferentes alternativas que se plantean teniendo en cuenta los objetivos a conseguir y las restricciones impuestas, en caso de riesgos el siguiente paso será la formulación de una estrategia efectiva en coste, después se revisarán los resultados del análisis de riesgos y finalmente se planificará la fase posterior. Una vez finalizado el primer ciclo se dará inicio al siguiente.

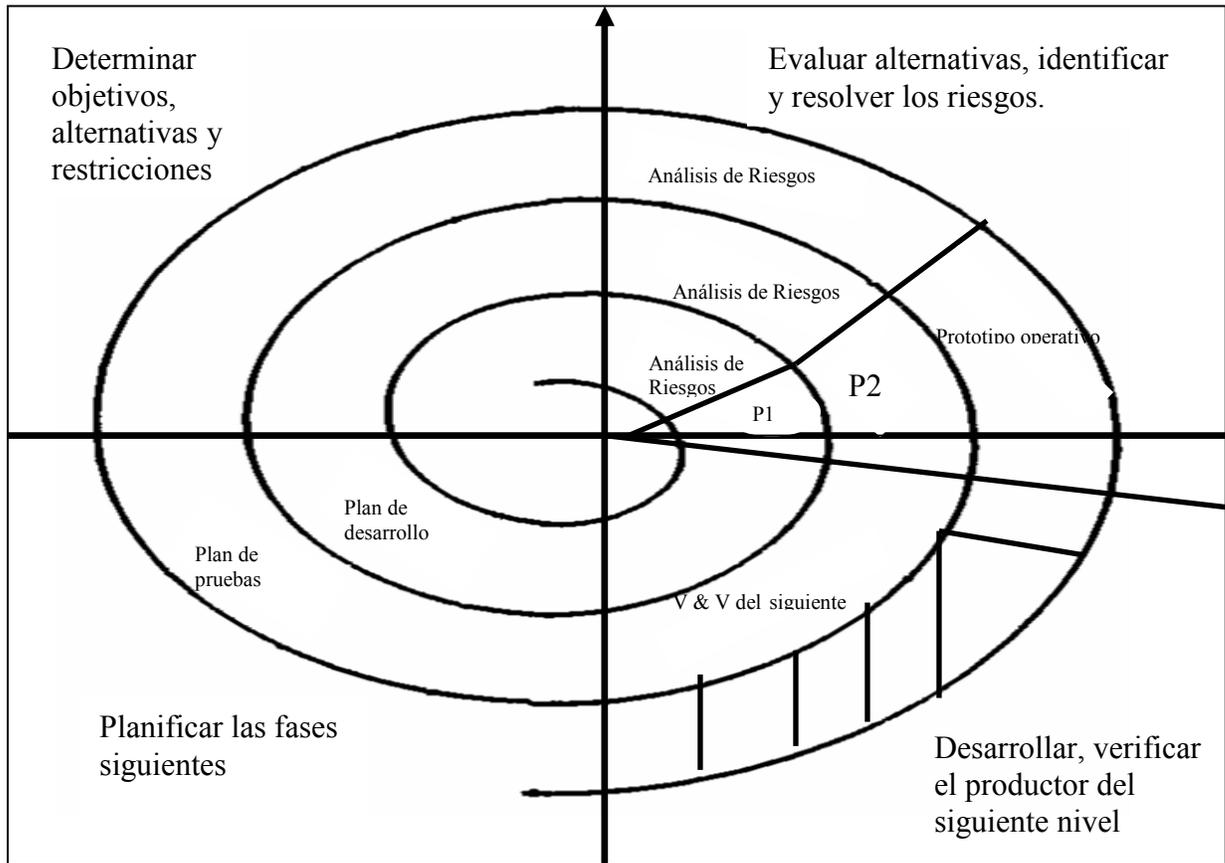


Fig. 2.3 Modelo en espiral

El modelo en espiral es el más actual y también uno de los más utilizados por los ingenieros de software, dentro de sus características principales destacan las que a continuación se mencionan:

- Existe un reconocimiento de las diferentes alternativas para alcanzar los objetivos de un proyecto.
- Esta basado en la identificación de riesgos y en las diferentes maneras de resolverlos.
- La división de los proyectos en ciclos.
- El modelo se adapta a cualquier tipo de actividad.

2.2 Arquitectura del software

La definición oficial de la arquitectura del software es la IEEE Std 1471-2000 que dice así: “La arquitectura del software es la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución”.

En un sentido amplio se entiende que la arquitectura del software es el diseño de más alto nivel de la estructura de un sistema, o aplicación y tiene la responsabilidad de:

- Definir los módulos principales.
- Definir las responsabilidades que tendrá cada uno de estos módulos.
- Definir la interacción que existirá entre dichos módulos:
 - ✓ Control y flujo de datos.
 - ✓ Secuenciación de la información.
 - ✓ Protocolos de interacción y comunicación.
 - ✓ Ubicación en el hardware.

La arquitectura del software de un sistema es el resultado de combinar decisiones técnicas, sociales y del negocio. Los ingenieros de software deben convencerse que su trabajo está influenciado por el negocio y el entorno social que los rodea.

Existen diversos modelos para desarrollar la arquitectura del software, los más comunes son:

- Vista Conceptual.

- Vista Lógica.
- Vista Física.
- Vista de Implementación.

Arquitectura cliente-servidor

Antes de entender este concepto es necesario definirlo por partes, analizando la definición de servidor y cliente en el área de informática.

Cliente

Tiene una relación muy estrecha con el servidor, por lo que cualquier ordenador que realice una petición o requiera un servicio por parte de otro ordenador que tenga características de servidor, será considerado como cliente.

Hablar de una arquitectura cliente-servidor es hacer énfasis de la estructura que tiene esta aplicación, normalmente en Internet se usa la arquitectura cliente-servidor, este tipo de organización se basa en que entre todos los ordenadores que están en la red, unos ofrecen servicios (mejor conocidos como servidores) y otros usan esos servicios (los llamados clientes) esto ocurre de manera frecuente cuando el usuario está visitando alguna página Web, en ese momento se está accediendo a un servicio que ofrece el servidor de páginas WEB (sirviéndole la página solicitada). Por lo tanto, el ordenador del usuario es un cliente y el que hospeda estas páginas es un servidor.

Servidor

Se entiende por servidor a la máquina u ordenador que mediante un software y una estructura física de red es capaz de suministrar información y servicios a otros ordenadores. Lo que hoy conocemos como Internet no es más que un conjunto de servidores configurados para brindar un servicio específico de

2.3 DFD (Diagrama de Flujo de Datos)

Los DFD no sólo se pueden utilizar para modelar aplicaciones de sistemas de procesos de información, sino también como un medio más óptimo para modelar organizaciones enteras, es decir, como una herramienta para la planeación estratégica y de negocios.

2.3.1 Definición de DFD

El diagrama de flujo de datos (DFD), es una herramienta que permite visualizar un sistema como una red de procesos funcionales, conectados entre sí por "conductos" y "tanques de almacenamiento" de datos. Es una de las herramientas más comúnmente utilizadas, sobre todo por sistemas operacionales en los cuales las funciones del sistema son de gran importancia y son más complejos que los datos que éste maneja.

- Proceso.
- Flujo.
- Almacén.
- Terminador.

Proceso

El primer componente del DFD se conoce como proceso. Los sinónimos comunes son burbuja, función, transformación. El proceso muestra una parte del sistema que transforma entradas en salidas. El proceso se representa gráficamente como un círculo y se nombra o describe con una sola palabra, frase u oración sencilla. Un buen nombre para un proceso generalmente consiste en una frase verbo-objeto tal como validar entradas o calcular

impuesto. En algunos casos, el proceso contendrá el nombre de una persona o un grupo (por ejemplo, un departamento o una división de una organización), o de una computadora o un aparato mecánico.

Flujo

Un flujo se representa gráficamente por medio de una flecha que entra o sale de un proceso; un ejemplo se muestra en la figura 2.6. El flujo se usa para describir el movimiento de bloques o paquetes de información de una parte del sistema a otra.



Fig. 2.5 Ejemplo de un flujo

En la mayoría de los sistemas los flujos realmente representan datos, es decir, bits, caracteres, mensajes, números de punto flotante y los diversos tipos de información con los que las computadoras pueden tratar. Nótese que el flujo de la figura 2.5 tiene nombre. El nombre representa el significado del paquete que se mueve a lo largo del flujo.

Los flujos muestran también la dirección: una cabeza de flecha en cualquier extremo (o posiblemente ambos) del flujo indica si los datos (o el material) se está moviendo hacia adentro o hacia fuera de un proceso (o ambas cosas). El flujo que se muestra en la figura 2.6.1(a), indica claramente que la información se está mandando hacia el proceso. Y el flujo denominado figura 2.6.2 (b) claramente indica que es una salida generada por el proceso, los datos que se mueven a lo largo de dicho flujo viajarán ya sea a otro proceso (como entrada)

o a un almacén o a un terminador. El flujo de dos cabezas que se muestra en la figura 2.6.3 (c) es un diálogo, es decir, un empaquetado conveniente de dos paquetes de datos (una pregunta y una respuesta) el mismo flujo. En el caso de un diálogo, los paquetes de cada extremo de la flecha deben nombrarse.

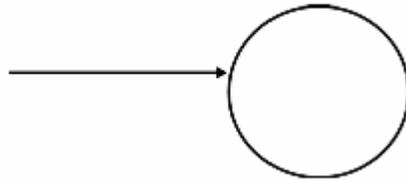


Fig. 2.6.1.a Flujo de entrada

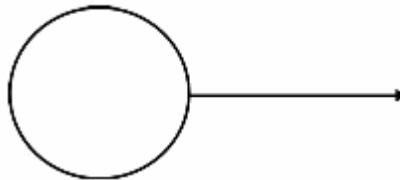


Fig. 2.6.2.b Flujo de salida

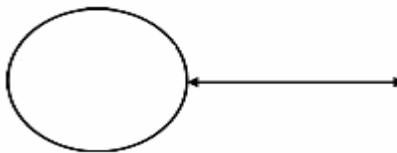


Fig. 2.6.3.c Flujo de diálogo

Almacén

El almacén se utiliza para modelar una colección de paquetes de datos en reposo. Se denota por dos líneas paralelas, como lo muestra la figura 2.7. De modo característico el nombre que se utiliza para identificar al almacén es el plural del que se utiliza para los paquetes que entran y salen del almacén por medio de flujos.

Base de Datos

Fig. 2.7 Representación gráfica de un almacén

Para el analista con conocimiento de proceso de datos es fácil referirse a los almacenes como archivos o base de datos; pero un almacén también puede consistir en datos almacenados en tarjetas perforadas, microfilm, microfichas, discos ópticos, etc. También puede ser un conjunto de fichas de papel en una caja de cartón, nombres y domicilios en un directorio, diversos archivos en un archivero, o varias formas no computarizadas.

Además de la forma física que toma el almacén, también existe la cuestión de su propósito, la base de datos existe como un área de almacenamiento diferida en el tiempo, necesaria entre dos procesos que ocurren en momentos diferentes.

Los almacenes se conectan por flujos a los procesos. Así, el contexto en el que se muestra en un DFD es uno de los siguientes (o ambos):

- Un flujo desde un almacén.
- Un flujo hacía un almacén.

Terminador

El terminador gráficamente se representa como un rectángulo, como se muestra en la figura 2.8. Los terminadores representan entidades externas con las cuales el sistema se comunica. Comúnmente, puede ser una persona, una organización externa o una agencia gubernamental, etc. Con la única condición de que se encuentre fuera del control del sistema que se está modelando. En algunos casos, un terminador puede ser otro sistema, como computacional con

el cual se este comunicando.

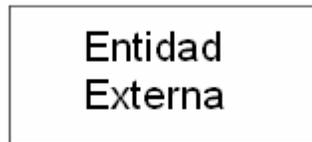


Fig. 2.8 Representación gráfica de un terminador

Existen tres cosas importantes que debemos recordar acerca de los terminadores:

- Son externos al sistema que se está modelando.
- Ni el analista ni el diseñador del sistema están en posibilidades de cambiar los contenidos de un terminador o la manera en que trabaja.
- Las relaciones que existan entre los terminadores no se muestran en el modelo de DFD.

En síntesis, el Diagrama de Flujo de Datos describe:

- Los lugares de origen y destino de los datos (los límites del sistema).
- Las transformaciones a las que son sometidos los datos (los procesos internos).
- Los lugares en los que se almacenan los datos dentro del sistema, y,
- Los canales por donde circulan los datos.

2.3.2 Características Principales de DFD

Características

- Se considera relevante porque posibilita comunicar diferentes modelos para así facilitar el entendimiento entre el usuario y el analista de sistemas.
- Es lógico porque no identifica soporte físico.
- Se construye en forma descendente, de lo general a lo particular.

Un DFD (Diagrama de Flujo de Datos) se profundiza expandiendo algunos de sus procesos en subprocesos. No hay un límite para el número de procesos.

Los diagramas derivados de los procesos principales se clasifican en niveles, los cuales son:

- Nivel 0: Diagrama de contexto.
- Nivel 1: Diagrama de nivel superior.
- Nivel 2: Diagrama de detalle o expansión

2.3.3 ¿Por qué Utilizar DFD?

El Diagrama de Flujo de Datos es muy práctico y sencillo de utilizar para modelar sistemas computacionales basados en programación estructurada utilizando símbolos gráficos, diccionarios de datos así como procesos y procedimientos. Además de que dentro de la programación estructurada se consideran los elementos básicos del sistema como Entrada-Proceso-Salida en función del software que se este utilizando, dicha metodología consta de 5 fases las cuales se desarrollaran a lo largo de este escrito y que son enunciadas a continuación:

- Análisis.
- Diseño.

- Codificación.
- Pruebas.
- Integración.

En la programación estructurada el diseño inicia una vez que ha culminado la fase de análisis de sistema. En este análisis se llega solo a la fase de integración y no toma en consideración los cambios que ocurren dentro del sistema en el proceso de análisis y diseño del sistema, es por ello que utilizar la herramienta de DFD es mucho más simple y demostrativo.

2.4 Sitios Web

La Web fue creada alrededor de 1990 por el inglés Tim Berners-Lee y el belga Robert Cailliau. La World Wide Web (o la "Web") es un sistema de documentos de hipertexto hospedados y enlazados a través de la gran red de redes mejor conocida como Internet, teniendo acceso a ellos con el uso de un navegador Web, dichas páginas pueden contener texto, imágenes o elementos multimedia.

Toda página Web tiene una estructura básica para ser implementada y requiere de las siguientes etapas:

- Prediseño.
- Diseño.
- Instalación y prueba.

Antes de empezar a diseñar una página Web se debe poner mucha atención en el contenido que se mostrará. Dicha información (texto, imágenes, vídeo, audio), requiere de un buen diseño para lograr la atención e interés del visitante, el utilizar un diseño simple es más agradable que una Web llena de

gifs, el éxito de una página se basa en elegir los colores adecuados, íconos, fotos, imágenes, etc. Esto hace más interesante visitar dicha página.

El uso de imágenes dentro del diseño de una página Web llama la atención del usuario que decida ingresar a ella, aunque esto puede afectar en gran parte la velocidad de acceso a la Web obligando a los usuarios sufrir largas esperas hasta que la información sea mostrada en la pantalla esto es un problema que esta fuera del alcance del diseño.

La creación de nuevos sistemas es una tarea que no podemos dejar al olvido, por lo que en los últimos tiempos se ha creado el sistema Web, pero lo primero es tener conocimientos suficientes de ello, siendo necesario investigar en que consiste un servidor Web así como su funcionamiento.

2.4.1 ¿Qué es un servidor Web?

El servidor Web es un programa que corre sobre el servidor que recibe las peticiones HTTP que le llegan y las atiende. Dependiendo del tipo de petición, el servidor Web ejecutará un programa para buscar una página Web, cabe aclarar que siempre devolverá algún tipo de resultado HTML al cliente que realizo la petición.

El servidor Web es un servidor que ofrece servicios dentro de la Word Wide Web en Internet y se considera como la parte esencial de un sitio Web ya que en el se ejecutarán todas las aplicaciones, el servidor Web es un programa encargado de ofrecer comunicación mediante un protocolo HTTP (Hypertext Transfer Protocol).

La codificación e interpretación de una página Web esta basada en el identificador de Recurso Uniforme (URI), es un sistema universal para referenciar recursos en la Web, así como en el Protocolo de Transferencia de Hipertexto (HTTP), que especifica cómo se comunican el navegador, el servidor, y el Lenguaje de Marcado de Hipertexto (HTML), usado para definir la estructura y contenido de documentos de hipertexto.

2.4.2 Protocolo HTTP

Se entiende por protocolo de comunicación al lenguaje o conjunto de reglas que permite el intercambio de información entre dos ordenadores. Existe una variedad de protocolos de comunicación y HTTP es uno de ellos. El protocolo de transferencia de hipertexto (HTTP, HyperText Transfer Protocol) es el protocolo usado para acceder a la Web (www) y es el encargado de procesar así como de dar respuestas a las peticiones para visualizar una página Web.

HTTP es un protocolo que no guarda ninguna información sobre conexiones anteriores. Al finalizar la transferencia de información, todos los datos se pierden. El HTTP esta basado en el modelo Cliente-Servidor, en donde un cliente HTTP inicia una conexión y realiza una solicitud al servidor. Éste responde a la petición con un recurso que puede ser un formulario, texto, gráfico o un mensaje de error, y finalmente se cierra la conexión.

2.4.2.1 Características principales del protocolo HTTP

Cada protocolo de comunicación tiene características particulares que hacen la diferencia entre los ya antes mencionados, por lo que es interesante tener conocimiento de ellas, entre las cuales se enuncian algunas de éstas:

- El protocolo HTTP generalmente utiliza el puerto 80.
- Su estructura es orientada a objetos.
- Permite que los sistemas no dependan del tipo de datos que se utilicen.
- Esta basado en un sistema de peticiones y respuestas.
- Se pueden realizar varias peticiones simultáneas por parte del cliente al servidor.
- Existen varias versiones referentes al protocolo HTTP como lo es la 0.9, 1.0, 1.1, etc.
- Realiza peticiones haciendo uso de métodos como GET, HEAD, POST, OPTIONS, etc.

2.4.3 Lenguaje HTML

Las páginas Web que son mostradas en la red se caracterizan por contener texto, imágenes, formularios, y objetos multimedia, pero sobre todo por que son hipertexto. Queda claro que las páginas no son elementos aislados, éstas se encuentra comunicadas mediante enlaces de hipertexto y es con la ayuda del lenguaje HTML (HiperText Markup Language), cuyas siglas significan "lenguaje hipertexto de marcas" , herramienta principal con la que se diseñan todos los elementos del www (world wide web) incluida la programación de páginas Web. Este lenguaje es el encargado de convertir un archivo de texto inicial en una página Web brindando la oportunidad de crear imágenes, animaciones, formularios, etc., logrando así que la página Web sea dinámica. Una de las características más importante de este lenguaje es que no es necesario ningún

programa especial para crear una página Web. Gracias a ello se ha conseguido que se puedan crear páginas con cualquier ordenador y sistema operativo.

El código HTML, esta basado en etiquetas de texto y por tanto lo único necesario para escribirlo es un editor de texto como los que se encuentran en cualquier sistema operativo: block de notas en Windows, UNIX, etc., también se pueden usar procesadores de texto, como lo son Microsoft Word ó WordPerfect pero hay que tener en cuenta que deberemos guardar el archivo en modo texto.

Un editor de texto simple, como el block de notas de Windows, es todo lo necesario para comenzar a crear una página Web. La estructura básica del lenguaje HTML es como la que se muestra a continuación:

```
<HTML>
<HEAD>
<TITLE> Mi Primer Página Web </TITLE>
</HEAD>
<BODY>
Mi Primera Página Web
</BODY>
</HTML>
```

Para poder interpretar el código HTML se requiere contar con un navegador (interprete de código HTML) instalado en el ordenador, éste software ayudará a visualizar las páginas Web. Hoy en día se cuenta con una gran variedad de navegadores pero de los más conocidos son el Internet Explorer y Netscape Navigator, dichos navegadores tendrán que ser compatibles con la versión de HTML que se este utilizando en ese momento con la intención de que se interpreten todos los atributos de la página.

2.4.3.1 Características Principales del Lenguaje HTML

- Es un lenguaje de hipertexto.
- Es el lenguaje estructural de una página Web.
- Esta compuesto por etiquetas.
- Es interpretado por un navegador.
- Existen varias versiones de este lenguaje como son la 3.0, 3.2, 4.0, etc.
- Se puede escribir en editores de texto simple cuidando guardar el archivo con la extensión HTML.
- Se puede escribir código en editores visuales como lo son; Macromedia Dreamweaver, Microsoft FrontPage, Adobe Pagemill, etc.

2.4.4 Lenguaje de programación PHP

Fue creado por Rasmus Lerdorf en 1995, inicialmente como scripts de Perl. Llamó a ese conjunto de scripts “Personal Home Page Tools” (PHP/FI), al pasar el tiempo se requería más funcionalidad, y Rasmus fue escribiendo una implementación en lenguaje C más completa, capaz de comunicarse con bases de datos, permitiendo a los usuarios desarrollar aplicaciones Web dinámicas.

PHP/FI, incluía algunas de las funciones básicas de PHP, tenía variables como las de Perl, interpretación automática de variables de formulario y sintaxis HTML. Por 1997, sale al mercado PHP/FI 2.0, la segunda versión de la implementación en C la cual fue sustituida en breve tiempo por las primeras versiones alfa de PHP 3.0, ésta se caracteriza por tener una sintaxis de lenguaje más potente y consistente además de estar orientada a objetos. Es hasta 1999 cuando se da paso a una nueva versión PHP 4.0 que es

actualmente la última versión liberada de PHP (Profesional Home Page), y con la cual se implementará este proyecto.

2.4.4.1 Características principales de PHP

- Es un lenguaje de alto nivel y se ejecuta en el servidor.
- Lenguaje orientado al desarrollo de aplicaciones Web.
- Hace uso excesivo de HTML.
- PHP puede hacer cualquier cosa que se pueda hacer con un Script CGI.
- Es rápido, a pesar de ser interpretado.
- Dispone de una gran cantidad de librerías.
- Es multiplataforma.
- Se programa utilizando objetos.
- Es un software libre.
- PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluyendo HP-UX, Solaris y OpenBSD), Microsoft Windows, etc.
- Soporta la mayoría de servidores Web.
- Con PHP tiene la libertad de elegir el sistema operativo y el servidor de su gusto.
- Tiene soporte para una gran cantidad de bases de datos.

2.4.4.2 Ventajas de utilizar PHP

La principal ventaja es que, al ejecutarse el código en el servidor, todas las páginas van a poder ser vistas en cualquier ordenador, independientemente del navegador que se tenga instalado.

Se trata de un lenguaje de programación gratuito y, por lo tanto, todo el mundo puede utilizarlo sin ningún coste, frente a otros lenguajes cuyo software es necesario comprar para su utilización. PHP puede procesar la información de formularios, generar páginas con contenidos dinámicos, o enviar y recibir cookies además de que su programación orientada a objetos, ahorra grandes cantidades de código acortando los tiempos al momento de programar. En la siguiente figura se puede ver el proceso que se realiza a la hora de visitar una página PHP:

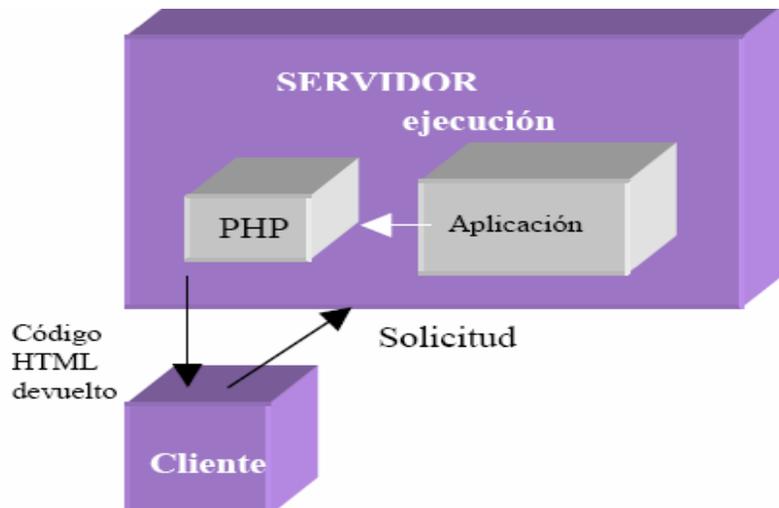


Fig. 2.9 Funcionamiento de una aplicación PHP

2.4.5 Características de JavaScript

Es un lenguaje de programación no tan complicado que por su flexibilidad y escasez de reglas se le considera un lenguaje script o lenguaje de archivos de comandos, es importante entender que un script es un tipo de lenguaje de programación sencillo dotado de pocas reglas, permite agregar pequeñas unidades funcionales a las aplicaciones, simplificar y automatizar la ejecución de las funciones en sí.

Algunas de las características más destacadas de JavaScript se enuncian a continuación:

- Amplía las capacidades de los documentos HTML.
- Está basado en objetos.
- Está controlado por eventos.
- JavaScript es seguro.
- Es independiente de la plataforma.
- JavaScript no es Java.

2.4.5.1 Diferencias entre Java y JavaScript

JavaScript	Java
➤ Se interpreta por el cliente.	➤ Se compila por el programador y se ejecuta por el cliente.
➤ El código está incrustado en HTML.	➤ El código (applet) se carga como módulo.
➤ Utilización sencilla de datos/tipos de datos.	➤ Reglas muy estrictas para los tipos de datos.
➤ Código sólo utilizable en documentos HTML.	➤ Permite la creación de programas independientes.
➤ Colabora con elementos HTML.	➤ Supera las capacidades de HTML.
➤ Posibilita el acceso directo a los objetos del navegador.	➤ Carece de acceso a los objetos del navegador y a su funcionalidad.
➤ Basado en objetos: se pueden utilizar los objetos	➤ Orientado a objetos: los applets constan de clases con

integrados, pero carece de creación de clases y de herencia.

herencia.

- Vínculo dinámico: las referencias de objetos se verifican en tiempo de ejecución.
- Vínculo estático: las referencias de objetos tienen que estar establecidas ya al efectuarse la compilación.

2.4.6 Manejadores de bases de datos

El sistema manejador de bases de datos se puede definir como la porción más importante del software de un sistema de base de datos. Un DBMS (DataBase Management System) es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica, el DBMS es conocido también como Gestor de Base de datos. Es el responsable de la comunicación (envío, actualización, recepción de datos) con la base de datos.

Las funciones principales de un DBMS son:

- Crear y organizar la base de datos.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser accedidos rápidamente.
- Manejar los datos de acuerdo a las peticiones de los usuarios.
- Registrar el uso de las bases de datos.
- Interacción con el manejador de archivos. Esto a través de las sentencias en DML al comando del sistema de archivos. Así el Manejador de base de datos es el responsable del verdadero almacenamiento de los datos.

- Respaldo y recuperación. Consiste en contar con mecanismos implantados que permitan la recuperación fácilmente de los datos en caso de ocurrir fallas en el sistema de base de datos.
- Control de concurrencia. Consiste en controlar la interacción entre los usuarios concurrentes para no afectar la inconsistencia de los datos.
- Seguridad e integridad. Consiste en contar con mecanismos que permitan el control de la consistencia de los datos evitando que estos se vean perjudicados por cambios no autorizados o previstos.

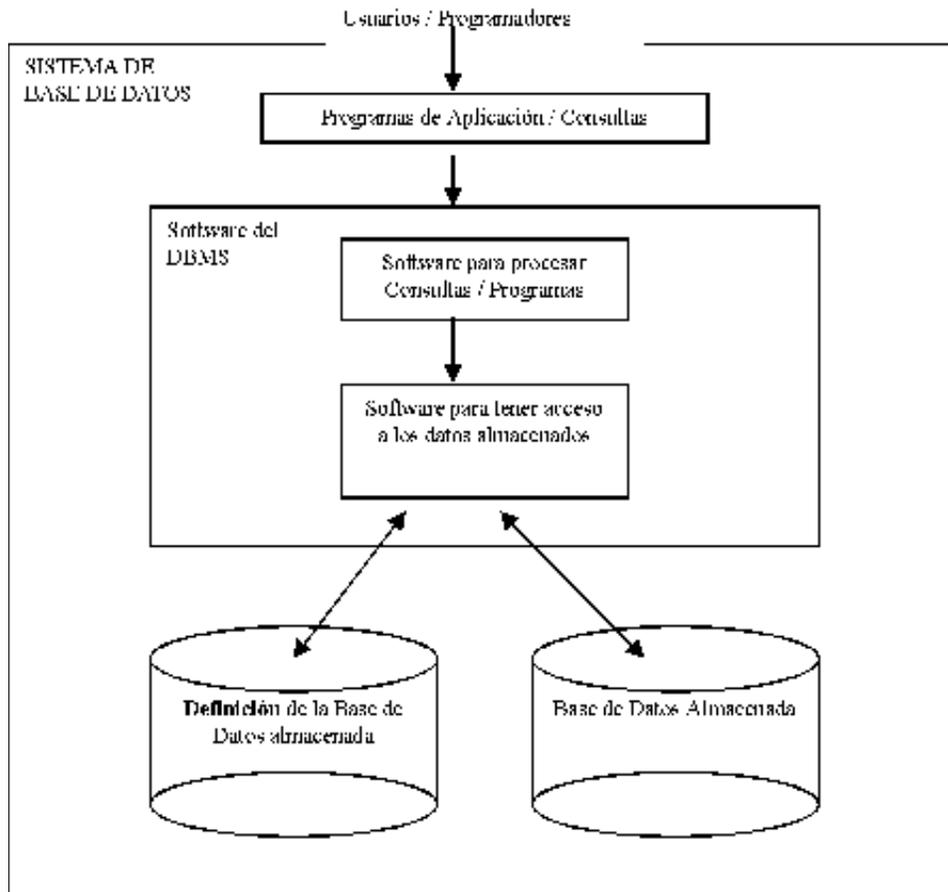


Fig. 2.10 Esquema de un DBMS

El DBMS interpreta las peticiones de entrada/salida del usuario y las manda al sistema operativo para la transferencia de datos entre la unidad de memoria secundaria y la memoria principal.

2.4.6.1 Diagrama entidad-relación

Conocido comúnmente por sus siglas como: E-R; Este modelo representa a la realidad a través de un esquema gráfico empleando la terminología de entidades, que son objetos que existen y son los elementos principales que se identifican en el problema a resolver realizando un diagrama y se distinguen de otros por sus características particulares denominadas atributos, el enlace que rige la unión de las entidades esta representada por la relación del modelo.

Un diagrama E-R puede ser representado también a través de una colección de tablas. Para cada una de las entidades y relaciones existe una tabla única a la que se le asigna como nombre el del conjunto de entidades y de las relaciones respectivamente, cada tabla tiene un número de columnas que son definidas por la cantidad de atributos y las cuales tienen el nombre del atributo.

2.4.6.1.1 Elementos de un diagrama entidad-relación

- Entidad.- Representado por un rectángulo y considerado el elemento principal de un diagrama E-R, dicho elemento es una representación del mundo real y sobre el que queremos almacenar información (Ejemplo: un carro). Las entidades están compuestas de *atributos* que son los datos que definen el objeto (para la entidad persona serían DNI, nombre, apellidos, dirección...). De entre los atributos habrá uno o un conjunto de

ellos que no se repite; a este atributo o conjunto de atributos se le llama *clave* de la entidad, (para la entidad persona una clave sería DNI). En toda entidad siempre hay al menos una clave que en el peor de los casos estará formada por todos los atributos de la tabla. Ya que pueden haber varias claves y necesitamos elegir una, lo haremos atendiendo a estas normas:

- ✓ Que sea única.
 - ✓ Que se tenga pleno conocimiento de ella.- ¿Por qué en las empresas se asigna a cada cliente un número de cliente?.
 - ✓ Que sea mínima, ya que será muy utilizada por el gestor de base de datos.
- Relación.- Se puede definir como una asociación entre entidades, sin existencia propia en el mundo real que estamos modelando, pero necesaria para reflejar las interacciones existentes entre entidades. Las relaciones pueden ser de tres tipos:
- ✓ Relaciones 1-1.- Las entidades que intervienen en la relación se asocian una a una.
 - ✓ Relaciones 1-n. - Una ocurrencia de una entidad está asociada con muchas (n) de otra.
 - ✓ Relaciones n-n.-Cada ocurrencia, en cualquiera de las dos entidades de la relación, puede estar asociada con muchas (n) de la otra y viceversa.

2.4.6.1.2 Representación gráfica de entidades y relaciones

Para asimilar fácilmente un diseño de datos cuando se emplea el modelo E/R se utilizan los siguientes elementos gráficos:

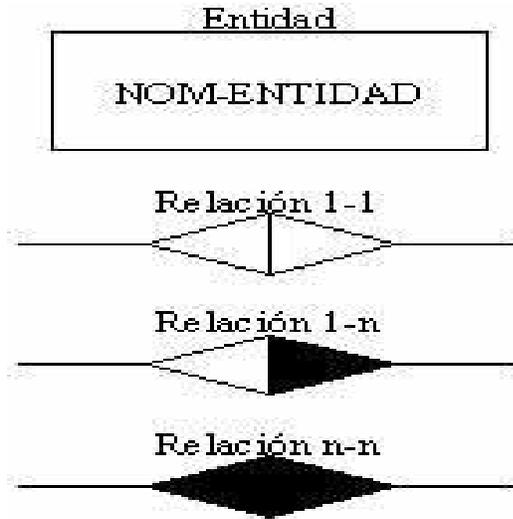


Fig. 2.11 Tipos de relaciones

Recordemos que un rectángulo nos representa a las entidades; una elipse a los atributos de las entidades, y una etiqueta dentro de un rombo nos indica la relación que existe entre las entidades, destacando con líneas las uniones de estas y que la llave primaria de una entidad es aquel atributo que se encuentra subrayado.

2.4.7 MySQL

MySQL surgió como un intento de conectar el gestor mSQL a las tablas propias de MySQL AB, usando sus propias rutinas a bajo nivel. Tras unas primeras pruebas, vieron que mSQL no era lo bastante flexible para lo que se necesitaba hacer, por lo que tuvieron que desarrollar nuevas funciones. Esto resultó

implementando una interfaz SQL a su base de datos, siendo totalmente compatible con mSQL.

Hasta el momento no se sabe con certeza de donde proviene su nombre. Por un lado dicen que sus librerías han llevado el prefijo 'my' durante los diez últimos años. Por otro lado, la hija de uno de los desarrolladores se llama My. No saben cuál de estas dos causas (aunque bien podrían tratarse de la misma), han dado lugar al nombre de este conocido gestor de bases de datos.

El software MySQL tiene licencia dual, pudiéndose usar de forma gratuita bajo licencia GNU o bien adquiriendo licencias comerciales de MySQL AB.

2.4.7.1 ¿Qué es MySQL?

MySQL es el servidor de bases de datos relacionales más utilizado por los programadores, utiliza múltiples tablas para almacenar y organizar la información. MySQL es una implementación Cliente-Servidor que consta de un servidor y diferentes clientes (programas/librerías), dicho software puede agregar, acceder, y procesar datos grabados en una base de datos.

2.4.7.2 ¿Porque usar MySQL?

MySQL es muy confiable, rápido, robusto y fácil de usar, puede manejar grandes cantidades de información. La conectividad, velocidad y seguridad hacen de MySQL un gestor de base de datos muy conveniente para acceder a bases de datos en Internet.

MySQL es un software de código abierto esto quiere decir que es accesible para cualquiera, puede usarlo o modificarlo. Se puede descargar MySQL desde Internet y usarlo sin pagar nada, de esta manera es considerado como un software libre en donde cualquiera puede inclinarse a estudiar el código fuente y cambiarlo de acuerdo a sus necesidades. MySQL usa el GPL (GNU Licencia Publica General) para definir que podemos y no podemos hacer con el software en diferentes situaciones. La relación entre PHP y MySQL hace posible consultar una base de datos de forma sencilla y ello le confiere un alto potencial a la hora de generar páginas Web dinámicamente, tal y como se muestra en la siguiente figura:

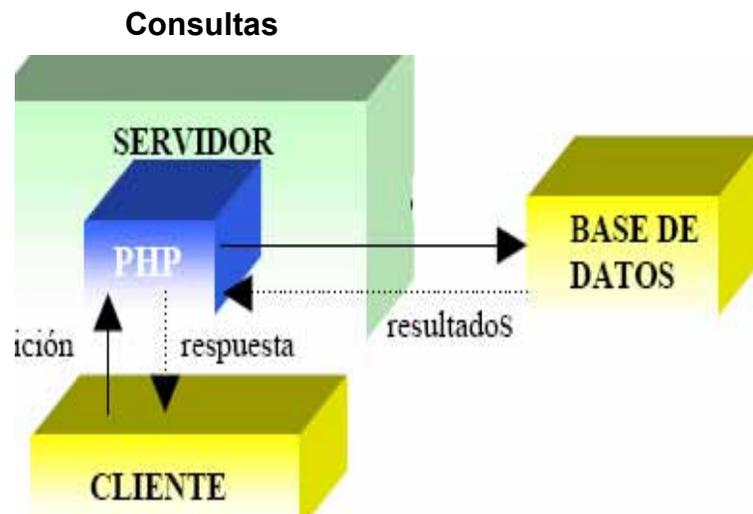


Fig. 2.12 Esquema del proceso de ejecución de una página Web con PHP

2.4.7.3 Principales características de MySQL

- Es portable.
- Robusto y muy rápido.

- Es un software libre.
- Multiproceso, es decir puede usar varias CPU si éstas están disponibles.
- Puede trabajar en distintas plataformas y Sistemas Operativos.
- Cuenta con sistema de contraseñas y privilegios muy flexibles además de ser seguros.
- Todas las palabras de paso viajan encriptadas en la red.
- Los clientes usan TCP o UNIX Socket para conectarse al servidor.
- El servidor soporta mensajes de error en distintas lenguas.
- Existe ODBC para utilizarse con Microsoft ACCESS.

Capítulo 3

Identificación y contexto del sistema

3.1 Antecedentes

Actualmente la aplicación de exámenes en línea es una tecnología poco utilizada debido a que, principalmente, no existe manera de garantizar que al aplicar un examen a través de medios electrónicos a distancia los alumnos lo resuelvan con ética. Por tal motivo, tienen que ser presentados por los alumnos bajo algún tipo de supervisión; esto porque generalmente se aplica una sola versión del examen a aplicar, pues generar diferentes versiones es un proceso que podría convertirse en una tarea complicada y/o tardada. Por otro lado, el tiempo de cada profesor muchas veces no es suficiente para aplicar y calificar todos los exámenes de su grupo, considerando que en la mayoría de las ocasiones, además de los exámenes, también tiene que evaluar proyectos y tareas que se hayan asignado durante el curso.

Este proyecto consiste en un sistema que busca automatizar al máximo las tareas que se necesitan para, citando las principales actividades: diseñar, aplicar y proporcionar el resultado de exámenes. Además, no sólo se desea automatizar las tareas antes mencionadas sino que se pretende se desarrollen a través de interfaces “amigables” para los usuarios, contemplando que muchos de los usuarios no están familiarizados, o bien, no tienen mucha experiencia con el uso de herramientas informáticas.

La meta de este trabajo de tesis es ofrecer un medio para facilitar la evaluación del desempeño académico de los alumnos, en primera instancia, de la ENP, plantel 7 “Ezequiel A. Chávez”; pero considerando que puede hacerse

extensivo a prácticamente cualquier plantel de nivel medio superior y, ¿porqué no?, a otros niveles de educación, aspirando a esto realizando pocas modificaciones a la solución desarrollada.

3.2 Planteamiento del problema

Dentro de la Dirección General de la Escuela Nacional Preparatoria (DGENP) de la UNAM, y debido al gran número de estudiantes que están en la institución, existe la necesidad de crear un sistema informático que agilice el proceso de evaluación. El problema se presenta con la forma en que actualmente se lleva a cabo todo el proceso, desde diseñar, formular y obtener los exámenes a aplicar, hasta pasar por el proceso de calificarlos, hacer llegar los resultados a las instancias correspondientes y, finalmente, el asentamiento de las calificaciones en actas.

Todos los trámites para lograr lo antes mencionado, como es de imaginar, requiere de mucho tiempo, recursos materiales, y del trabajo de varias personas. Así entonces, se busca simplificar todas las tareas que forman parte del proceso de evaluación, y resulta verdaderamente importante lograrlo si consideramos que no sólo sería útil para la preparatoria número 7, sino para todas las dependencias de la UNAM a nivel medio superior, considerando que las modificaciones requeridas deberían ser mínimas de un plantel a otro si se sabe que tienen la misma estructura organizacional.

Hoy en día el uso de las herramientas virtuales da la posibilidad de que la formación académica sea autónoma y a distancia, es por ello que en Internet se pueden encontrar aplicaciones ya desarrolladas que funcionan como portales educativos, software interactivo y en algunos casos sistemas de auto evaluación, aclarando que la mayoría de ellas se han realizado con un fin

lucrativo. En cambio, la solución propuesta, cuyo nombre es “Sistema Integral para la Evaluación de Alumnos a Distancia (SIEAD)” será versátil, multidisciplinario y sencillo de utilizar para el usuario, además de ser una aplicación gratuita y para las dependencias educativas de la UNAM, en primera instancia, dirigida para el nivel medio superior.

Teniendo presente que la tarea principal es el desarrollo de un sistema informático, podemos decir que los objetivos que se buscan alcanzar con tal sistema son:

3.3 Objetivos generales

- Crear una herramienta que sea confiable, flexible, rápida y fácil de usar para el diseño, aplicación y calificación de exámenes.
- Un sistema que sirva de apoyo a los profesores para la evaluación de sus alumnos invirtiendo menos tiempo y esfuerzo.
- Llevar un registro de todas las operaciones realizadas en el sistema para utilizarlo en futuros estudios que sirvan en la toma de decisiones que mejoren el proceso de evaluación, y aún más, de la educación misma.

3.4 Objetivos particulares

- Elaborar exámenes de forma libre o automática.
- La creación y mantenimiento del banco de reactivos.
- La generación, presentación en pantalla, y resultado en tiempo real de los exámenes.

- Generar exámenes diferentes para cada alumno.
- Historial de los reactivos presentados por alumno, así como del orden en que se presentaron.
- Registro, consulta y actualización de todos los elementos necesarios para el diseño de los exámenes.
- Llevar un control de los usuarios y permisos de los mismos en el acceso y uso del sistema, con opción a modificarlos por quienes cuenten con los privilegios adecuados.
- Registro de todos los cambios que se efectúen en el sistema por usuario, fecha y hora en que se realicen.
- Proporcionar diferentes formas de mostrar el resultado de un examen tanto a alumnos como profesores.
- Reutilización de reactivos y exámenes completos.
- El uso compartido de reactivos y exámenes entre los profesores, es decir, la posibilidad de utilizar elementos que no hayan sido creados por ellos mismos.

3.5 Alcances y Limitaciones del Sistema

- El profesor debe proporcionar todas las preguntas al sistema en un formato definido a lo largo del desarrollo del proyecto.
- La lista de asignaturas, alumnos y profesores y usuarios del sistema los debe introducir a la base de datos un administrador a través de las interfaces correspondientes que forman parte del sistema.
- El sistema no pretende evaluar el aprovechamiento de los alumnos, sino más bien, ser un medio para agilizar tal proceso dando un resultado medible.
- El sistema definirá el orden de las preguntas del examen y el de sus respuestas sin repetir dicho orden para los reactivos y/o sus respuestas.

Esto último depende básicamente del número de veces que se haya utilizado un reactivo en los exámenes de un alumno.

- Debido a la enorme complejidad que actualmente existe para reconocer los diferentes tipos de sintaxis y semántica a través de algoritmos computacionales, no se podrán aplicar preguntas abiertas ya que el sistema no podrá analizar si la respuesta es correcta, por lo que ésta tendrá que ser de opción múltiple.
- El tiempo necesario para realizar todos los procesos implicados en el diseño, aplicación y resultado de los exámenes será considerablemente rápido.
- Con base en la información que se registre en el sistema, existe la posibilidad de una futura creación de un módulo que despliegue resultados estadísticos de diferentes aspectos, como por ejemplo, la materia que se reprueba más, la que se más se aprueba, los reactivos más fáciles, los más difíciles, cuántos alumnos presentan exámenes y en qué asignaturas se distribuyen, cuantos usuarios hay registrados, etc. Todo esto para poder tomar decisiones oportunas que mejoraren la eficiencia del proceso enseñanza-aprendizaje.

3.6 Descripción de las alternativas de solución

Antes de entrar en detalle en cuestiones técnicas cabe aclarar que ya se han descrito en los capítulos anteriores los términos que se utilizarán en lo que resta del documento.

Ahora bien, de las metodologías de ingeniería de software descritas anteriormente, se ha decidido utilizar la metodología del modelo incremental, que llega a ser algo parecido al ciclo de vida en espiral y que en definitiva ambos son una variante del modelo de ciclo de vida clásico. La principal razón

de utilizar esa metodología es que nos permite tomar un número pequeño de variables o requerimientos y una vez analizados, diseñados, realizados, implantados y aprobados podemos seguir con otros requerimientos y así sucesivamente hasta tener terminada la aplicación; además de ser flexible y permitirnos regresar a un punto dado en el desarrollo para verificar o hacer las correcciones pertinentes.

Resulta muy importante hablar sobre el ambiente de desarrollo elegido para la creación del sistema: el ambiente Web. Este ambiente permite desarrollar desde simples contenidos estáticos, hasta sistemas completos que pueden realizar tareas tan complejas como cualquier sistema desarrollado en otros ambientes, con la principal ventaja de que los primeros, son razonablemente fáciles de desarrollar y que por naturaleza están orientados para su uso en red, desde una red interna o local hasta en la red más grande del mundo: Internet, sin la necesidad de utilizar librerías especiales para hacerlos funcionar en esa modalidad.

Lo mencionado arriba permite que los sistemas en ambiente Web sean accesibles a cualquier usuario de cualquier parte del mundo, además de ofrecer gran disponibilidad debido a que, de ser necesario, podrían funcionar todo el tiempo sin necesidad de consumir grandes recursos materiales, humanos ni económicos; todo lo anterior con un desempeño en tiempo muy aceptable. Esto resulta trascendental en la búsqueda de nuestro objetivo si decimos que un sistema que ayude en el proceso de evaluación involucra a muchos y diferentes usuarios que no se encuentran en un mismo lugar ni usan los mismos equipos.

Además de todas las ventajas que tienen los sistemas diseñados para la Web, tienen la capacidad de ser usados de forma local sin realizar alguna

configuración especial. Todo lo mencionado da como resultado un ambiente de desarrollo muy flexible.

Pasando a otro punto, el SIEAD se decidió desarrollarlo en el lenguaje de programación PHP, pues a pesar de existir otros lenguajes como java o alguna tecnología punto net de Microsoft, PHP es un lenguaje fácil de utilizar y gratuito, sin olvidar mencionar que es el lenguaje que cuenta con mayor número de proveedores de servicios de hospedaje para sistemas creados con esta tecnología. Aún más, PHP es un lenguaje muy portátil, multiplataforma, y que permite desarrollar sistemas muy confiables, eficientes y prácticamente sin limitaciones en el desarrollo de soluciones.

Para hacer las validaciones del lado del usuario se decidió utilizar el lenguaje JavaScript, pues es el lenguaje mayormente utilizado en el desarrollo de páginas Web, soportado por la mayoría de los navegadores de Internet. Además, es un lenguaje que forma parte del software libre, muy poderoso, robusto, y fácil de utilizar que permite hacer páginas Web más sencillas de usar para los usuarios, más dinámicas y, en general, obtener un mejor desempeño del sistema.

En cuanto al DBM a utilizar, se considera que MySQL es un sistema de gestión de bases de datos relacional que nos permite crear, consultar y administrar una base de datos de acorde con nuestras necesidades. Este gestor de bases de datos es probablemente el más usado en el mundo del software libre, debido a que es rápido, eficiente, y fácil de usar además, de contar con una amplia gama de librerías y funciones que permiten interactuar con varios lenguajes de programación, también resultan sencillo su instalación y configuración.

En lo que respecta al sistema operativo Linux en comparación con otros, como por ejemplo, cualquier versión de Windows de Microsoft, tiene mayor

estabilidad y seguridad, además de ser considerablemente más barato, y algunas distribuciones sin ningún costo. A pesar de que su uso es más complicado en el aspecto de no ser tan “amigable” con un usuario común, o que no sea tan gráfico como sucede en Windows; este punto no resultará trascendente si consideramos que su uso estará limitado únicamente al equipo que contendrá la aplicación desarrollada, y que el acceso a ella será vía Web; por lo que el usuario final del SIEAD sólo necesitará conocimientos básicos en computación, fundamentalmente: cómo acceder y navegar en una página Web.

Capítulo 4

Análisis del sistema

4.1 Análisis de los requerimientos del sistema

El análisis es la fase más importante de la ingeniería de software pues es aquí de donde se obtiene la información sobre lo que se va a realizar, con qué información se cuenta para realizarlo y qué mejoras se pueden incorporar en la solución a desarrollar. Si se hace primero el diseño o se desarrolla directamente se pueden provocar inconsistencias en el sistema, y se tendrían que hacer constantes modificaciones al código antes de obtener un resultado adecuado, lo que provoca pérdida de recursos, tiempo y esfuerzo.

El análisis de requerimientos es el proceso que ayuda a comprender hacia donde va el proyecto, por lo cual se deben realizar actividades como las que se describirán a continuación.

4.1.1 ¿Qué es lo que el usuario desea?

El Ing. Alejandro Villagómez, el cual en calidad de profesor en la institución que solicita la realización de este proyecto, hizo de nuestro conocimiento la necesidad que existe de crear una herramienta para formular, aplicar y calificar exámenes de manera automática reduciendo el tiempo que el profesor emplea actualmente en ese proceso. También se comentó que dicho sistema debe ser fácil de utilizar, rápido, seguro y sobre todo confiable, pues un error en cualquier parte del proceso podría provocar grandes inconvenientes.

También se desea que al sistema se tenga acceso a través de Internet, específicamente vía Web, pues de esta manera todos los usuarios involucrados en las actividades que cubrirá el sistema podrán ingresar al mismo con gran facilidad sin requerir de un software especial que no fuera un navegador común.

Cómo parte de la funcionalidad del sistema se desea que éste tenga un orden lógico, simple, apegado lo más posible a lo que es un proceso tradicional de aplicación de exámenes, pero aprovechando las ventajas que dan los elementos informáticos disponibles con las herramientas de desarrollo, como por ejemplo, las interfaces gráficas, el registro de los movimientos realizados en el sistema, la historia detallada de cada alumno en el uso del sistema, la posibilidad de dar un resultado al instante, entre otros.

Además, otro punto muy importante que se persigue con el desarrollo del sistema es crear un banco de información que en el futuro permita explotar todos los datos que se registren a través de la aplicación para tomar decisiones en cualquier aspecto del proceso de evaluación: desde la formulación de reactivos, el contenido de los temarios, etc., hasta en la forma en que se realiza actualmente dicho proceso. Todo esto con parámetros medibles y que el sistema podrá proporcionar con relativa facilidad.

4.1.2 Clasificación de los requerimientos

4.1.2.1 Requerimientos del entorno

El sistema usa el entorno y lo necesita como una fuente de los servicios necesarios para que funcione. Ejemplos del entorno podemos mencionar: sistemas operativos, sistema de archivos, bases de datos. Los requerimientos mínimos de este tipo para que el sistema funcione son:

- Un navegador de Internet, por ejemplo Mozilla o Internet Explorer.
- Un servidor web, en nuestro caso “Apache”.
- Un manejador de base de datos, el SIEAD utilizará MySQL.
- Conexión a Internet.

4.1.2.2 Requerimientos de interfaz

Una interfaz es el medio de interacción entre un sistema y el ser humano u otros sistemas; es la especificación formal de como el sistema recibe los datos o los manda al exterior. Usualmente se especifica el protocolo, el tipo de información, el medio para comunicarse y el formato de los datos que se van a comunicar.

Para el SIEAD se utilizará el protocolo HTTP, el tipo de información transmitida serán básicamente archivos con formato HTML y peticiones o resultados de una consulta a la base de datos. El medio de comunicación será Internet y el formato de los datos será texto plano.

4.1.2.3 Requerimientos funcionales

Tal como lo dice su nombre son los que describen lo que el sistema debe de hacer. Es importante que se describa el qué y no el cómo, éstos requerimientos

conforme avanza el proyecto se convierten en los algoritmos o lógica, al final son gran parte del código del sistema.

Como ya se comentó, el sistema deberá funcionar en Web y almacenar en una base de datos los registros de los alumnos; tanto datos personales como el historial de los exámenes que haya presentado y sus respectivas calificaciones. Esto servirá para que, cuando un alumno presente un nuevo examen, con base a sus datos registrados en el sistema se genere dicho examen evitando que tengan el mismo orden los reactivos (y/o respuestas de tales reactivos) que hayan sido incluidos, si es el caso, en un examen anterior del alumno en cuestión. Por otro lado, el sistema sólo permitirá realizar un examen a la vez, esto iniciar significa que un alumno no podrá más de una sesión en el sistema.

El sistema deberá tener todas la interfaces necesarias para ingresar y actualizar los datos necesarios en el proceso de diseño, aplicación y calificación de los exámenes. Esto involucra el registro y mantenimiento de: colegios, materias (y sus respectivos temarios), reactivos, usuarios, exámenes (estructura, validación, programación, etc.).

Se debe conseguir la integridad y coherencia de los datos para evitar que, por ejemplo, un examen presentado por un alumno se pueda consultar tal y como lo vio el usuario en el momento de contestarlo aún si los datos del sistema sufrieran algún cambio, pues si no fuera así, la utilidad de una herramienta como la que se busca realizar carecería prácticamente toda utilidad o ventaja.

4.1.2.4 Requerimientos de eficiencia

Estos requerimientos nos informan las características de desempeño y eficiencia que debe tener el sistema. Debido a que es un sistema en tiempo

real, se tiene que considerar que al menos se deben tener 60 ó más sesiones simultáneas. La base de datos, a su vez, debe soportar al menos el mismo número de conexiones; cabe aclarar que esto último depende principalmente de la velocidad de respuesta del servidor y del tráfico que pueda existir en la red.

El sistema debe tener la capacidad de operar todo el tiempo y deberá ser adaptable a los posibles cambios tecnológicos, con la finalidad de lograr un ciclo de vida mayor para el sistema.

Antes de poder utilizar esta aplicación se deberá entregar un manual de usuario a los individuos que harán uso del sistema para que aprendan el funcionamiento y alcances del mismo.

Otro aspecto a considerar, son las restricciones que deberá tener el sistema. Estas son:

- Sólo un usuario con los privilegios adecuados, puede proporcionar las preguntas (y modificarlas si fuera necesario) que utilizará el sistema para generar exámenes.
- La fecha y hora de aplicación de un examen sólo podrá ser establecida por el usuario que lo diseñó.
- Un alumno podrá consultar únicamente sus propias calificaciones, sólo el profesor podrá ver los resultados de todos los alumnos que realizaron el examen programado por él.
- Únicamente podrán hacer uso del sistema usuarios registrados.

4.2 Identificación de los tipos de usuarios que harán uso del sistema

La aplicación desarrollada está orientada al uso académico, por lo tanto, como primer usuario tenemos al alumno que será el que con menos privilegios cuente pues sólo tendrá acceso a los exámenes que le correspondan. Por otro lado, tenemos al profesor quien es el encargado de la creación y programación de dichos exámenes; teniendo disponibles los procesos implicados: alta y mantenimiento de reactivos, temarios, materias entre otros.

Con base en el párrafo anterior, es indispensable tener a un usuario administrador el cual se encargará de dar mantenimiento al sistema en caso de alguna contingencia, actualizar la base de datos cada vez que sea requerido y verificar el funcionamiento de la aplicación. Por lo tanto, el administrador no tendrá ninguna restricción sobre el uso de la aplicación. En conclusión se tendrán tres tipos de usuarios:

- Administrador
- Profesor
- Alumno

4.3 Estudio de los riesgos

Es necesario analizar los riesgos que se corren durante el proceso correspondiente a la realización del examen vía Web. Todo sistema tiene sus ventajas y desventajas por lo que se pretende analizar de una manera minuciosa objetiva este proceso, con la meta de que el sistema sea altamente eficiente y seguro para la obtención de resultados óptimos.

Para ello cabe aclarar que la realización del examen será dentro de los laboratorios de informática pertenecientes a las instalaciones de la UNAM, inicialmente, el plantel número siete de la ENP. En la sala de cómputo estará presente un asesor que indique paso a paso la realización del mismo y que a su vez verifique los datos personales de cada alumno, comprobando que su número de cuenta, y la foto de su credencial correspondan al alumno que se dispondrá a realizar el examen. Tener estas medidas preventivas de seguridad ayudara a obtener resultados confiables y precisos sobre el avance académico de los alumnos; aunado a ello se debe considerar que dicho sistema será instalado en un servidor de plataforma Linux lo que dará mayor seguridad y confiabilidad en el resguardo de la base de datos, eso contribuirá al esfuerzo de evitar que el sistema sea “hackeado” o pueda presentarse alguna falla en el funcionamiento del sistema debido a agentes externos.

Todo alumno que desee hacer uso del sistema deberá ingresar su nombre de usuario (el cuál será su número de cuenta) y password, de lo contrario se le negará el acceso al sistema. Sólo el administrador del sistema podrá realizar modificaciones en él, en cuanto a los tutores, se les otorgará ciertos permisos para el control de reactivos. El sistema le dará oportunidad al profesor para incluir nuevos reactivos en la base de datos, sin embargo, debe considerarse que el profesor es responsable de no duplicar reactivos, pues el sistema sólo mostrará los reactivos existentes, pero no podrá determinar casos de redundancia de información. Es este un punto en el que se debe poner especial énfasis, pues sino se toman medidas apropiadas para evitar la redundancia de información, esto puede afectar tanto al desempeño del sistema como a su utilidad en el proceso de evaluación.

Este análisis es realizado suponiendo escenarios de trabajo aún inexistentes pero que puedan ser creados posiblemente. Dicho análisis es de gran ayuda

para prever errores futuros. Este análisis es resultado del análisis de requerimientos, ya que en éste se definieron las características generales de la aplicación.

- Como primer riesgo tenemos que las preguntas no sean suficientes para aplicar exámenes diferentes y/o mostrar las respuestas en diferente orden. Para evitar que se presente esta situación es recomendable que el sistema tenga la capacidad de crear exámenes con un orden de preguntas y respuestas diferentes entre sí, a pesar de que sean las mismas preguntas, y solicitar al profesor verificar exista un número mínimo de reactivos.
- Otro riesgo es que una pregunta no tenga asociada una respuesta correcta. Por lo que resulta importante validar la indicación de las respuestas correctas antes de que se guarde la pregunta en la base de datos.
- Se debe considerar también que en un momento dado no se puedan soportar suficientes sesiones simultáneas en el sistema. Este punto depende en gran parte de los recursos disponibles del servidor, por lo que se debe hacer un estudio de la demanda esperada de los usuarios y de las características físicas mínimas que debe tener la computadora que alojará el sistema.
- Que un alumno intente presentar dos o más exámenes al mismo tiempo.
- Riesgo en la coherencia de los datos, evitar que cambios en los datos del sistema modifiquen exámenes ya aplicados.

4.4 Etapas de desarrollo del sistema

El sistema a desarrollar se ha dividido en tres grandes módulos o secciones, pues son en torno a estas que recaen todas las actividades contempladas en el desarrollo. A continuación se enuncian y explican cada uno de ellos.

- Módulo de diseño.
- Módulo de aplicación.
- Módulo de administración.

A partir de estos módulos se analizan y proponen los campos y tablas necesarias para el diseño del sistema. En esta sección se plantea la estructura de los registros de las tablas a utilizar, estando conscientes de que todo sistema está propenso a una gran variedad de cambios frecuentes con la única intención de hacerlo mas funcional y eficiente.

4.4.1 Módulo de diseño

En este módulo, previa autenticación del profesor, se mostrará en pantalla las asignaturas a las que tiene acceso, el tipo de materia depende del colegio al que pertenezca, observe la tabla compuesta de los siguientes campos:

Tabla de profesores

LP	PP	NP	APP	AMP
(login del profesor)	(password del profesor)	(nombre del profesor)	(apellido paterno del profesor)	(apellido materno del profesor)
CP	TU	EP	MP	UA
(correo del profesor)	(tipo de Usuario)	(estatus del profesor)	(materias del profesor)	(usuario de alta)
FA				
(fecha de alta)				

Para poder mostrar la pantalla que contendrá las asignaturas que forman parte del colegio al que pertenece el profesor, y entendiendo que se permitirá elegir sólo una de ellas, para poder agregar, modificar o eliminar algún reactivo de dicha asignatura, necesitamos contar con una tabla compuesta de los siguientes campos:

Tabla de Asignaturas

CA	VA	DA	CA	EA
(clave de asignatura)	(versión de asignatura)	(descripción de asignatura)	(colegio de asignatura)	(estatus de asignatura)
UAA	FAA			
(usuario de alta de asignatura)	(fecha de alta de asignatura)			

En el campo DA (Descripción de la asignatura) se incluirán todos los títulos de las unidades que conforman a la correspondiente asignatura, introduciendo un carácter especial (como por ejemplo, \$) que nos permitirá identificar el término y/o el inicio del título de una unidad.

4.4.1.1 Proceso para agregar, modificar o eliminar un reactivo

El proceso para agregar, modificar o eliminar un reactivo es el siguiente: Una vez elegida la asignatura se mostrarán las unidades que pertenecen a dicha materia, así como el número de reactivos registrados para cada unidad. Así, una vez elegida la unidad en la que se trabajará, se irá presentando un reactivo a la vez para el caso de eliminar o modificar reactivo, para la opción de agregar reactivo, se pasaría al formato en blanco para tal fin.

Para lo anterior, haremos uso de una tabla como la siguiente:

Tabla de reactivos

IR	VR	IMR	ITR	VTR
(identificador del reactivo)	(versión del reactivo)	(identificador de la Materia a la cual pertenece el reactivo)	(identificador del tema al cual pertenece el reactivo)	(versión del tema al cual pertenece el reactivo)
TR	NDR	OAR	OBR	OCR
(texto del reactivo)	(nivel de dificultad del reactivo)	(opción A del reactivo)	(opción B del reactivo)	(opción C del reactivo)
ODR	OER	JAR	JBR	JCR
(opción D del reactivo)	(opción E del reactivo)	(justificación A del reactivo)	(justificación B del reactivo)	(justificación C del reactivo)
JDR	JER	OCR	UA	FA
(justificación D del reactivo)	(justificación E del reactivo)	(opción correcta del reactivo)	(usuario alta)	(fecha alta)

Cabe aclarar que el número absoluto del reactivo IR (identificador del reactivo), será un dato implícito, es decir, no se pedirá nunca que se dé tal dato, pues habrá un proceso que asignará tal número, correspondiendo dicho número a una numeración consecutiva, es decir, a la suma total de reactivos registrados en la tabla Reactivos y la tabla indicada abajo. Así entonces, es importante mencionar que una vez dado de alta un reactivo, el IR(identificador del reactivo) que tenía asignado, no será utilizado para algún otro reactivo en el futuro, esto también se debe a que el reactivo que se dé de baja, pasará a otra tabla que se mostrará debajo de éste párrafo; algo similar a lo que sucederá cuando se modifique un reactivo, sólo que en éste último caso, la versión más actual tendrá un IR (identificador del reactivo) nuevo que se guardara en el campo VR(versión del reactivo), y se almacenará en la tabla como si se tratase de un nuevo reactivo, mientras que la versión anterior del mismo, conservará el IR

que le fue asignado al ser creado en un inicio, pero tal reactivo ahora de traspasará a la tabla que se muestra abajo.

Y todo esto, con la finalidad de llevar un registro de las operaciones realizadas en el sistema, así como poder recuperar siempre los reactivos que llegaron a ser utilizados por el sistema en algún examen.

Tabla de cambiosreactivos

IR	VR	IMR	ITR	VTR
(identificador del reactivo)	(versión del reactivo)	(identificador de la Materia a la cual pertenece el reactivo)	(identificador del tema al cual pertenece el reactivo)	(versión del tema al cual pertenece el reactivo)
TR	NDR	OAR	OBR	OCR
(texto del reactivo)	(nivel de dificultad del reactivo)	(opción A del reactivo)	(opción B del reactivo)	(opción C del reactivo)
ODR	OER	JAR	JBR	JCR
(opción D del reactivo)	(opción E del reactivo)	(justificación A del reactivo)	(justificación B del reactivo)	(justificación C del reactivo)
JDR	JER	OCR	UA	FAR
(justificación D del reactivo)	(justificación E del reactivo)	(opción correcta del reactivo)	(usuario alta)	(fecha de alta del reactivo)
T	UC	FC	E	C
tipo	Usuario Cambio	Fecha Cambio	enviado	contestado

El campo VR (versión del reactivo) tendrá el IR (identificador del reactivo) del reactivo que lo sustituyó en la tabla Reactivos, en caso de haberse tratado de una actualización del reactivo, así como la fecha y hora en que se dio tal cambio. Como último paso de este módulo, contaremos con la opción de salir del mismo.

4.4.2 Módulo de aplicación

Ingresar en un campo la contraseña del alumno que va a presentar el examen PU (password del usuario) para mostrar las asignaturas a las que tiene acceso, es decir, aquellas que no ha aprobado, esto último, como resultado de los siguientes pasos: primero, comparar el campo PU (password del usuario) con el campo LU (login del usuario) de una tabla como la que se muestra a continuación, para asegurarnos de que está inscrito.

Tabla de usuarios

LU (login del usuario)	V versión	PU (password del usuario)	NU (nombre del usuario)	APU (apellido paterno del usuario)
AMU (apellido materno del usuario)	CU (correo del usuario)	TU (tipo de Usuario)	EU (estatus del usuario)	MU (materias en las que esta inscrito el usuario)
UA (usuario alta)	FA (fecha alta)			

Y como segundo paso, una vez verificada la inscripción del alumno, buscar en una tabla que tendrá el registro de todos los exámenes aplicados, el login del alumno (LU), y si es encontrada en un registro, en dicho registro, ubicar el campo que contenga la clave de la asignatura y la calificación correspondiente, y si tiene una calificación aprobatoria en tal asignatura, la misma será eliminada de la lista que contiene todas las asignaturas que forman parte del plan de estudios. Así entonces, la tabla que llevará el registro de los exámenes aplicados será como sigue:

Tabla de exámenes

IE (Identificador del	IM (Identificador	TE (Titulo del	IR (Identificador de
---------------------------------	-----------------------------	--------------------------	--------------------------------

examen)	de la materia)	examen)	reactivos)
IVR	RE	UAE	FAE
(Identificador de Versiones de Reactivos)	(Realización del examen)	(Usuario dado de Alta en el examen)	(Fecha de Alta y hora del examen)

El campo RE (Realización del examen) contendrá datos formados por los campos IE (Identificador del examen), IR (Identificador de reactivos), y TE (Titulo del examen) de los reactivos utilizados en el examen, así como la opción elegida OCR (Opción Correcta del Examen) en el mismo para cada reactivo. La siguiente tabla registrará la hora en la cual se realizará el examen.

Tabla de bitácora

IE	IR	FE	DHE
(Identificador del examen)	(Identificador del reactivo)	(Fecha en que realizarán el examen)	(Duración en horas del examen)
DME	IV		
(Duración en minutos del examen)	(Identificador de validación)		

Y también mencionar que la calificación (C), la dará el sistema con base en comparar OCR (Opción Correcta del Examen) con la respuesta dada por el alumno en el proceso de aplicación del examen. Así como el sistema insertará la fecha, hora y el tiempo que durará el examen.

Una vez elegida la materia en la que se presentará el examen, éste se generará de manera pseudo-aleatoria, es decir, se tomarán las preguntas de la tabla Reactivos que cumplan con las especificaciones dadas con anticipación, como serían: número de reactivos de los que se compondrá el examen, las unidades que se evaluarán de la asignatura, y obviamente, si es el caso en que el alumno haya presentado un examen de la asignatura anteriormente, los

reactivos utilizados en el examen anterior, no podrán ser utilizados en el nuevo examen (éste último paso, es el sistema que consultando la tabla Exámenes, buscará los reactivos del pasado examen). A la vez que se van seleccionando los reactivos que cumplen con las especificaciones antes mencionadas, se irá llenando una tabla que almacenará temporalmente el examen que se aplicará al alumno, en este caso, es una tabla compuesta de los campos:

Tabla de reactivosExamen

IE	IR	VR	IVT	UA
(identificador del examen)	(identificador del reactivo)	(versión del reactivo)	(identificador de la versión de los temas)	(usuario dado de alta en el sistema)
FA				
(fecha de alta en el sistema)				

Así entonces, cuando se haya terminado de generar el examen, se irán presentando los reactivos al alumno uno a uno, con opción a regresar o avanzar al reactivo anterior o siguiente, según sea el caso, sin embargo, se dará un determinado tiempo para concluir el examen, en caso de no terminar en el tiempo asignado, se dará por terminado el examen, y se evaluará con lo que se tenga hasta ese momento.

La calificación que resulte de la evaluación del examen será presentada al alumno, y si se da el caso de no haber aprobado, se le mostrará también un diagnóstico indicando en qué reactivos se equivocó, dando una justificación del porque no es la respuesta correcta. Como último paso de este modulo, contaremos con la opción de salir del mismo.

4.4.3 Módulo de administración

Ingresar en un campo la clave del administrador (CAD) para mostrar un menú con las opciones con que se administrará el sistema, al cual se tendrá acceso dependiendo del resultado de comparar el campo clave del administrador con el campo CAD de una tabla compuesta de los siguientes campos:

Tabla de administradores

CAD	NAD	APAD	AMAD
(clave del administrador)	(nombre del administrador)	(apellido paterno)	(apellido materno)

Las opciones del menú mencionado serán las siguientes:

- **Reactivos:** en esta opción, se podrán realizar los procesos de Altas, Cambios y Bajas de reactivos. Así como la consulta de los reactivos que han sido dados de baja o han sido modificados (dejando abierta la posibilidad de hacerlo de manera general, es decir, todos los casos que se han dado, o bien, consultando por fechas, asignaturas, unidades de una asignatura, entre otras formas de consulta)
- **Exámenes:** en esta opción se podrán consultar los exámenes realizados, sin embargo, no se podrá modificar información alguna. Siendo posible también realizar consultas por fechas, asignaturas, alumnos, etc.
- **Claves:** en esta opción, se podrá registrar en el sistema a profesores para que puedan realizar los procedimientos propios de los reactivos; así como alumnos para que puedan presentar un examen e incluso, administradores. Lo anterior, por medio de asignarles sus respectivas contraseñas o claves de acceso, o si es el caso, modificarles la misma.

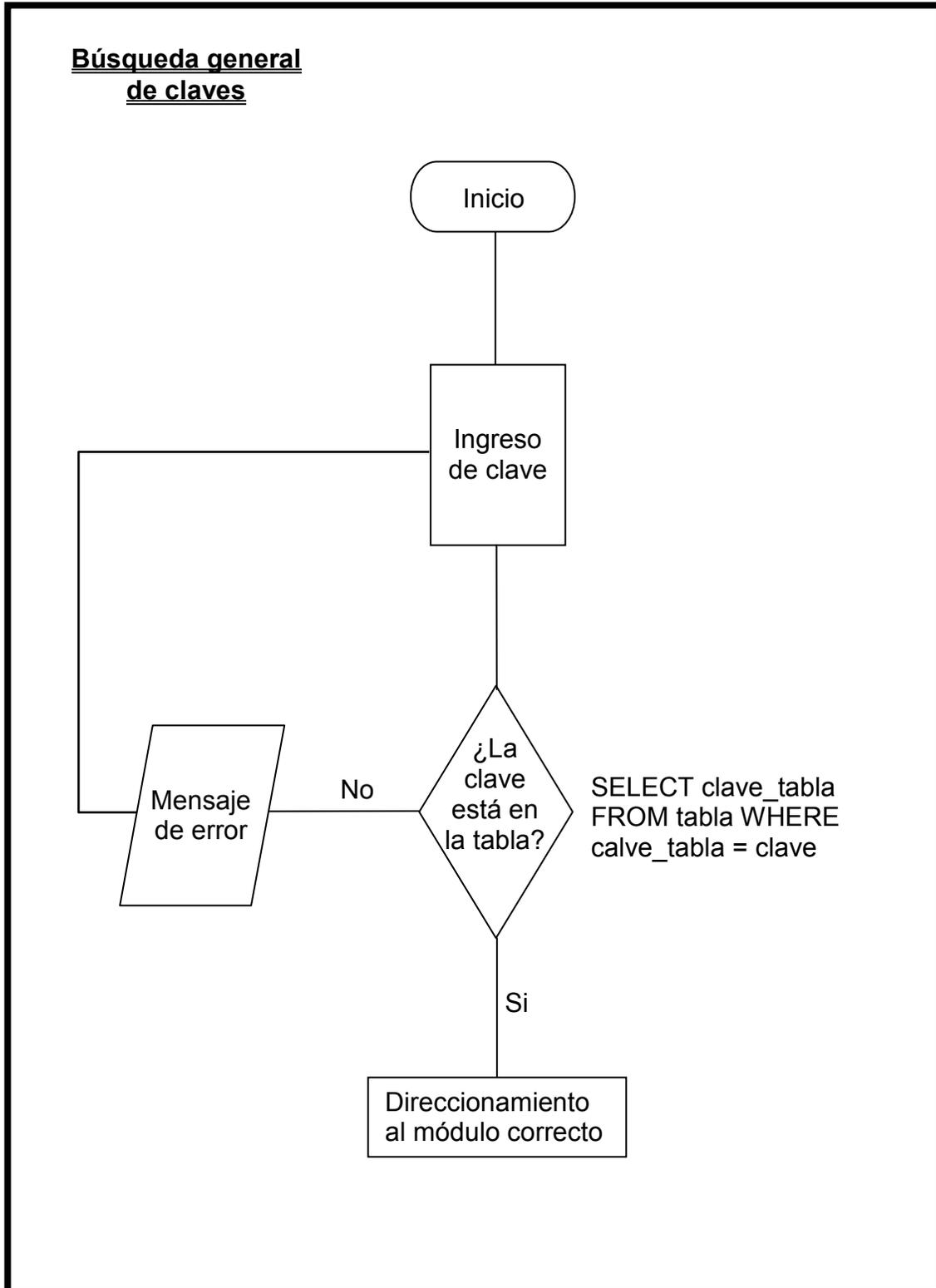


Fig. 4.1 Búsqueda general de claves

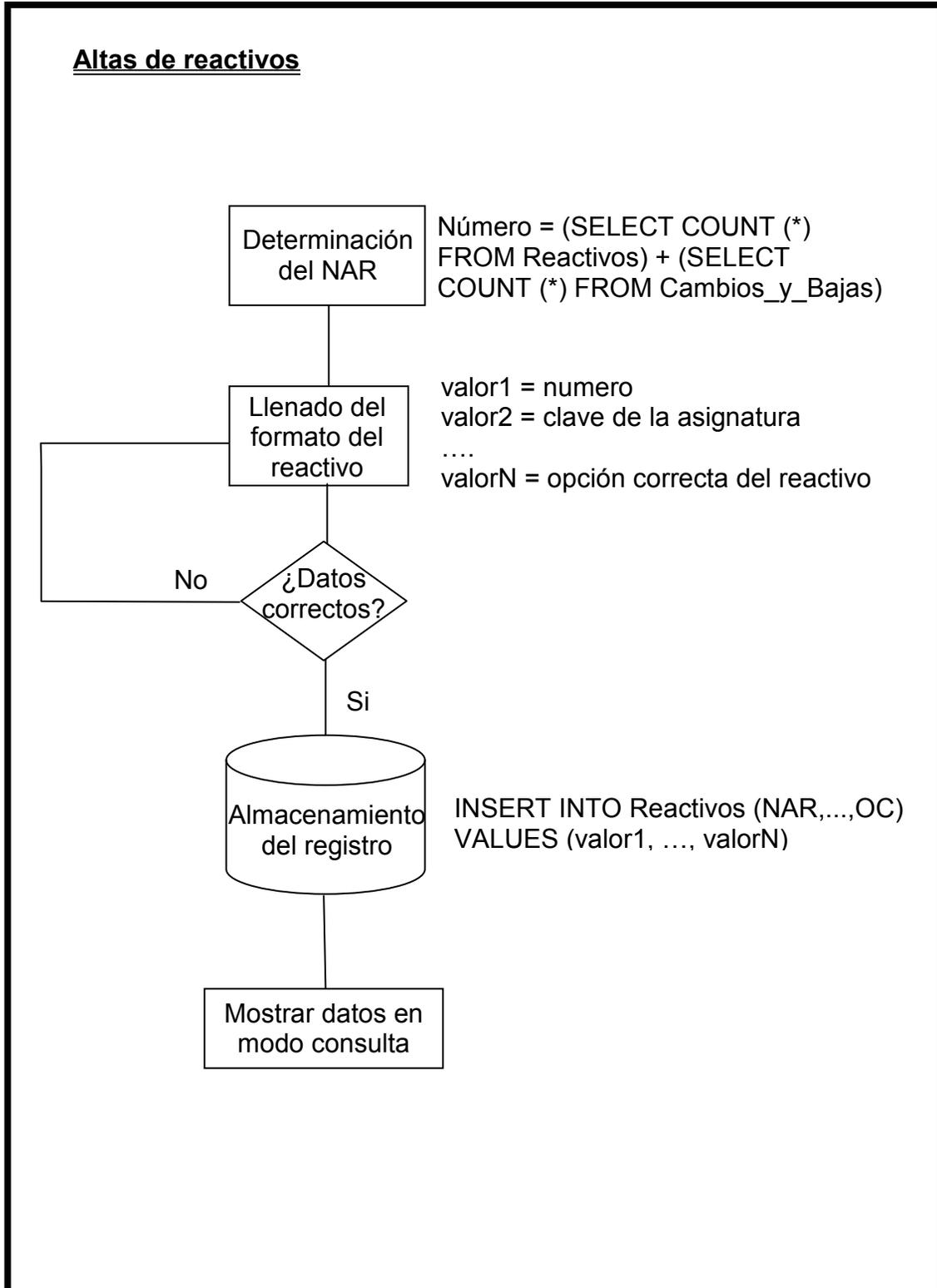


Fig. 4.2 Alta de reactivos

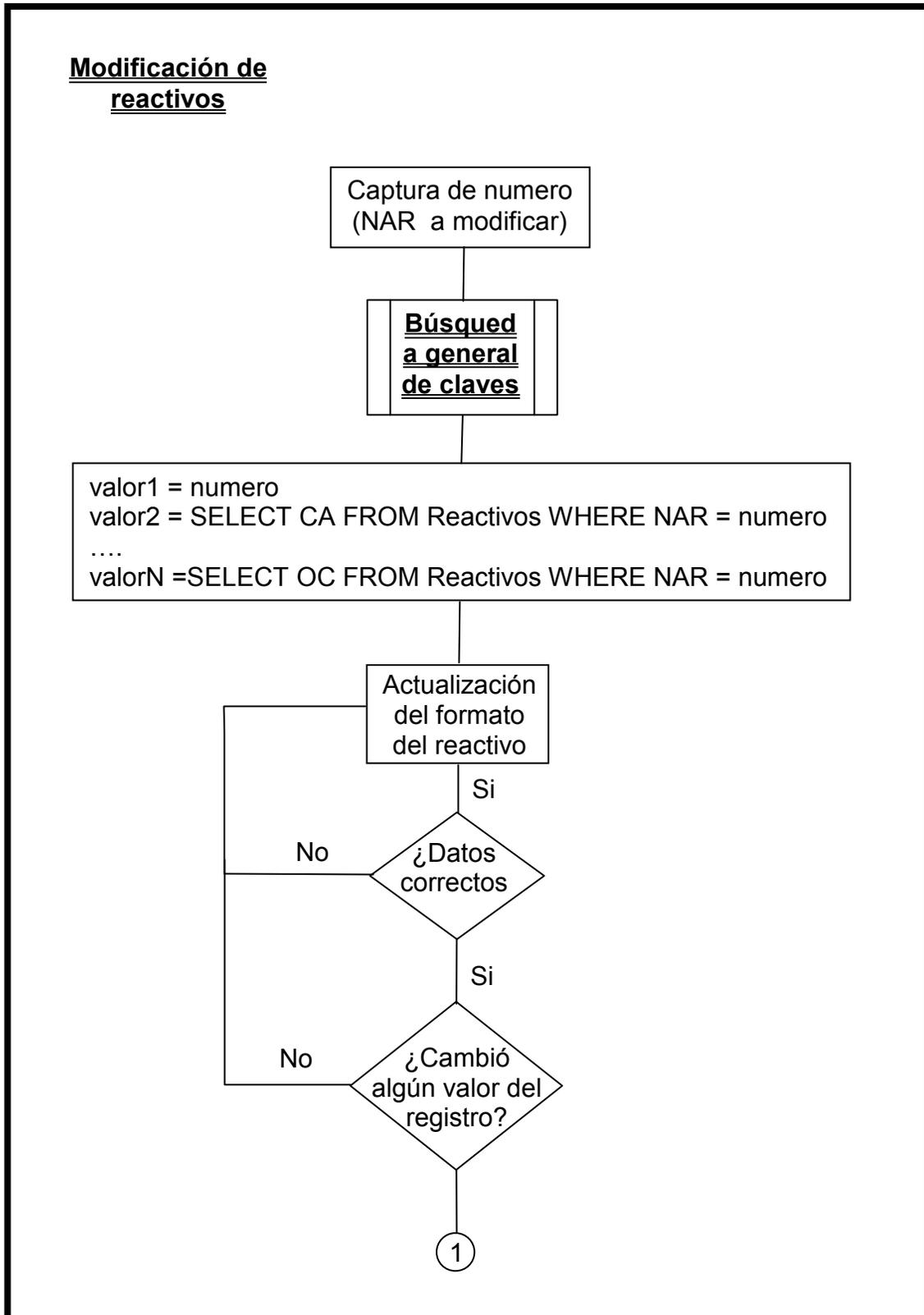


Fig. 4.3.a Modificación de reactivos

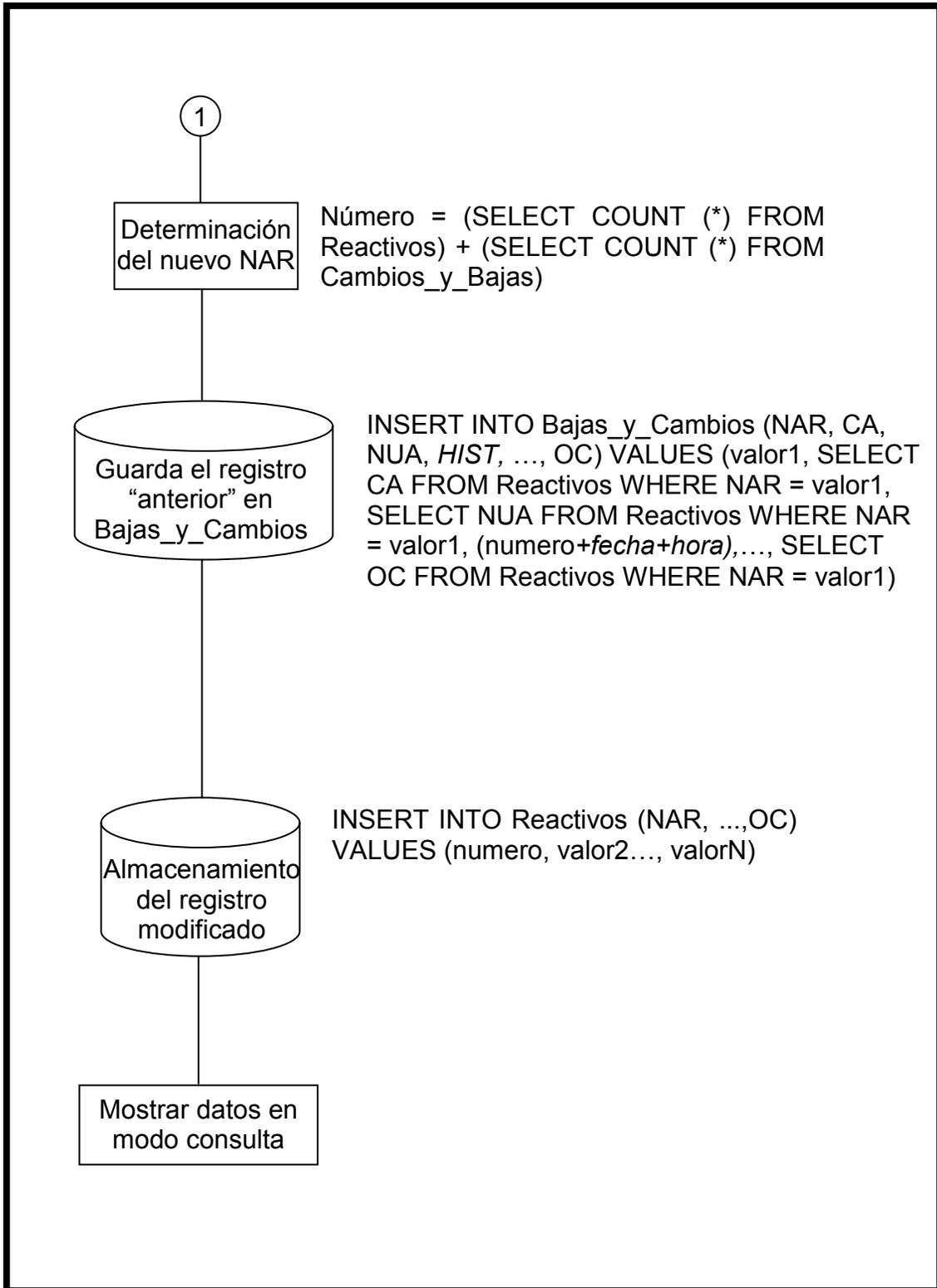


Fig. 4.3.b Modificación de reactivos

Capítulo 5

Diseño del sistema

En este capítulo se aplica la segunda fase de la ingeniería del software: el diseño. Este no puede ser desarrollado sin antes haber completado la fase de análisis ya que están totalmente ligados. Es decir, si se realiza un buen análisis se tiene una buena base para obtener un diseño correcto, el cual consiste en llevar al siguiente paso el análisis. El diseño es lo más cercano a la realidad aunque sigue en papel, pues es aquí en donde se definen las tecnologías informáticas a utilizar y de que medios se pueden valer, teniendo en cuenta el proceso que se definió en la primera fase.

5.1 Diseño e implementación de diagramas de flujo de datos

Como ya se ha mencionado en capítulos anteriores DFD (Diagrama de Flujo de Datos) es una herramienta que permite modelar, construir y documentar los elementos que forman un sistema basado en programación estructurada.

5.1.1 Diagrama de flujo de datos

El diagrama de flujo de datos (DFD), es una herramienta que permite visualizar un sistema como una red de procesos funcionales, conectados entre sí por "conductos" y "tanques de almacenamiento" de datos. Siendo éste, una de las herramientas más comúnmente usadas en la programación estructurada.

Es importante recordar tal y como se menciona en el capítulo 2 de éste documento, que los DFD no sólo se pueden utilizar para modelar sistemas de

proceso de información, sino también como una manera de modelar organizaciones enteras, es decir, como una herramienta para la planeación estratégica y de negocios.

5.1.2 Diagramas de flujos de datos aplicados al sistema

Los diagramas de flujo de datos son parte del análisis debido a que ayudan a describir que es lo que el sistema debe realizar. Es decir, describen un uso del sistema y como interactúa con el usuario, tal y como se muestra en el siguiente diagrama:

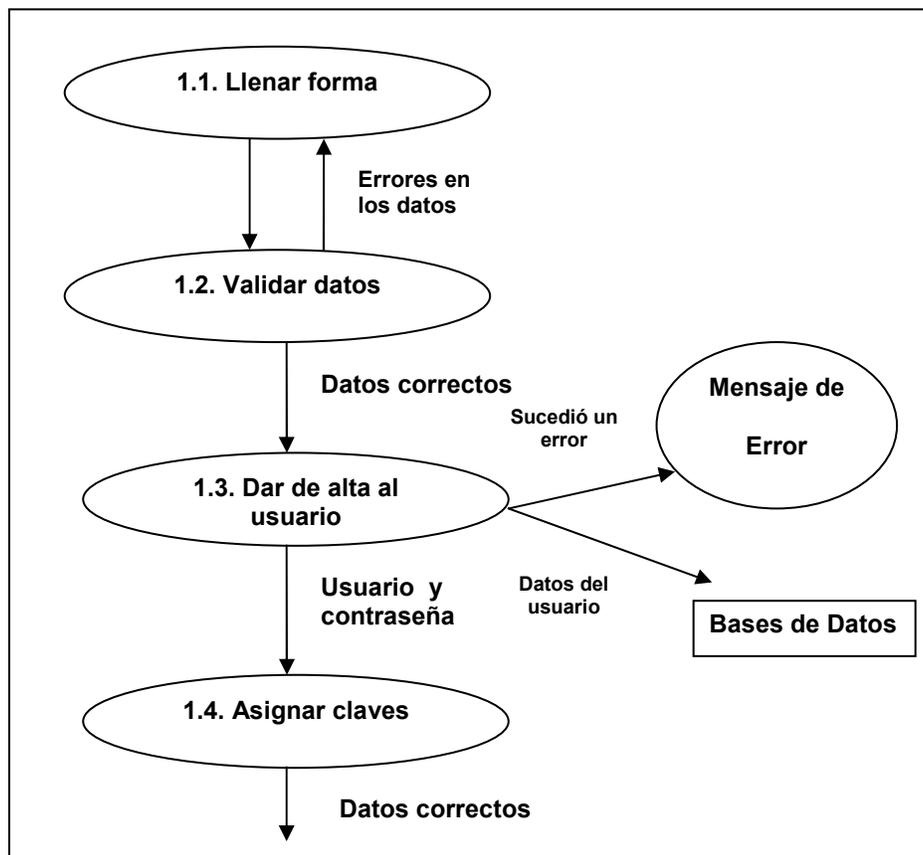


Fig. 5.1 DFD para registro de usuarios, Nivel 1

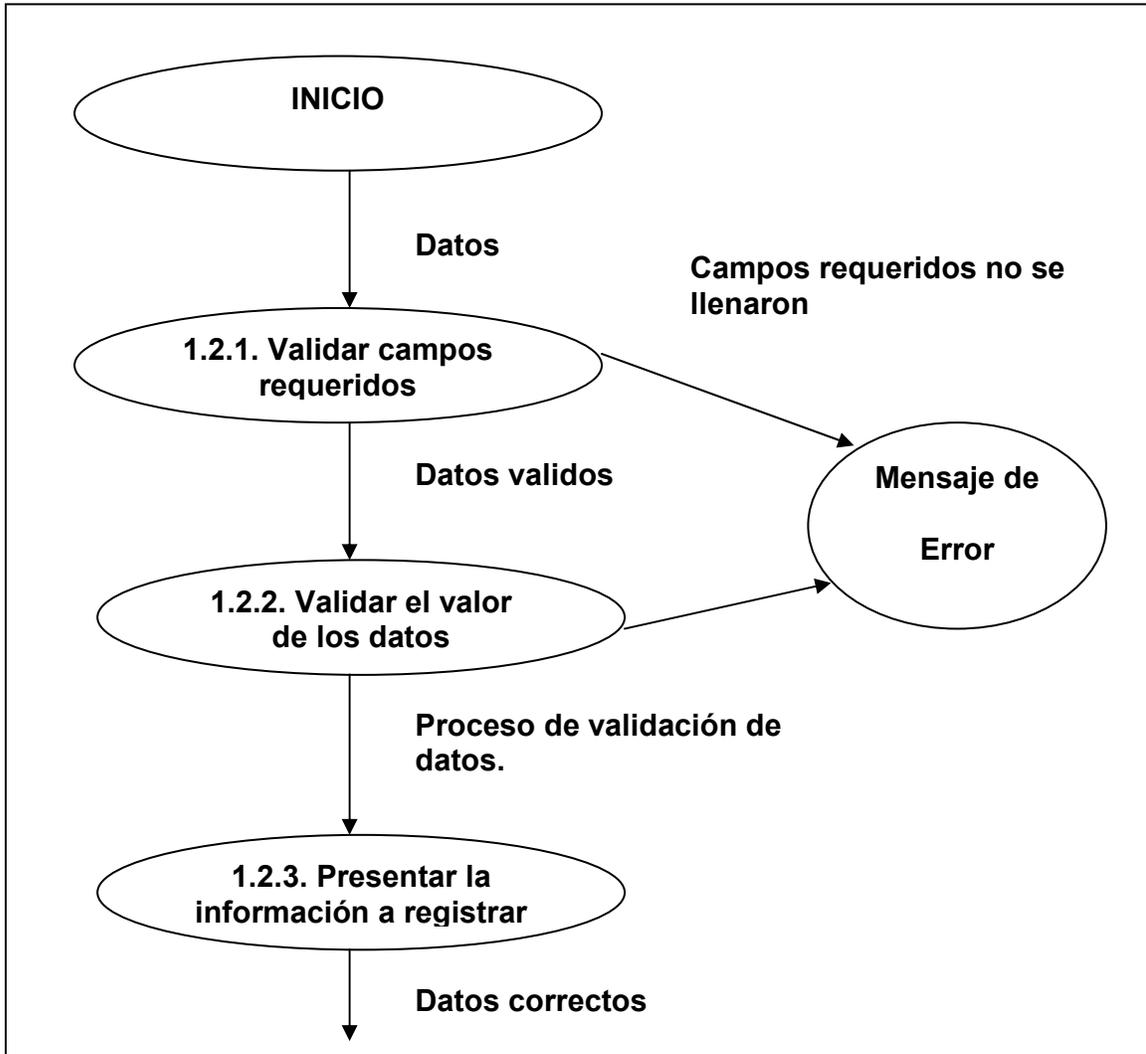


Fig. 5.2 DFD para validar datos, Nivel 1.2

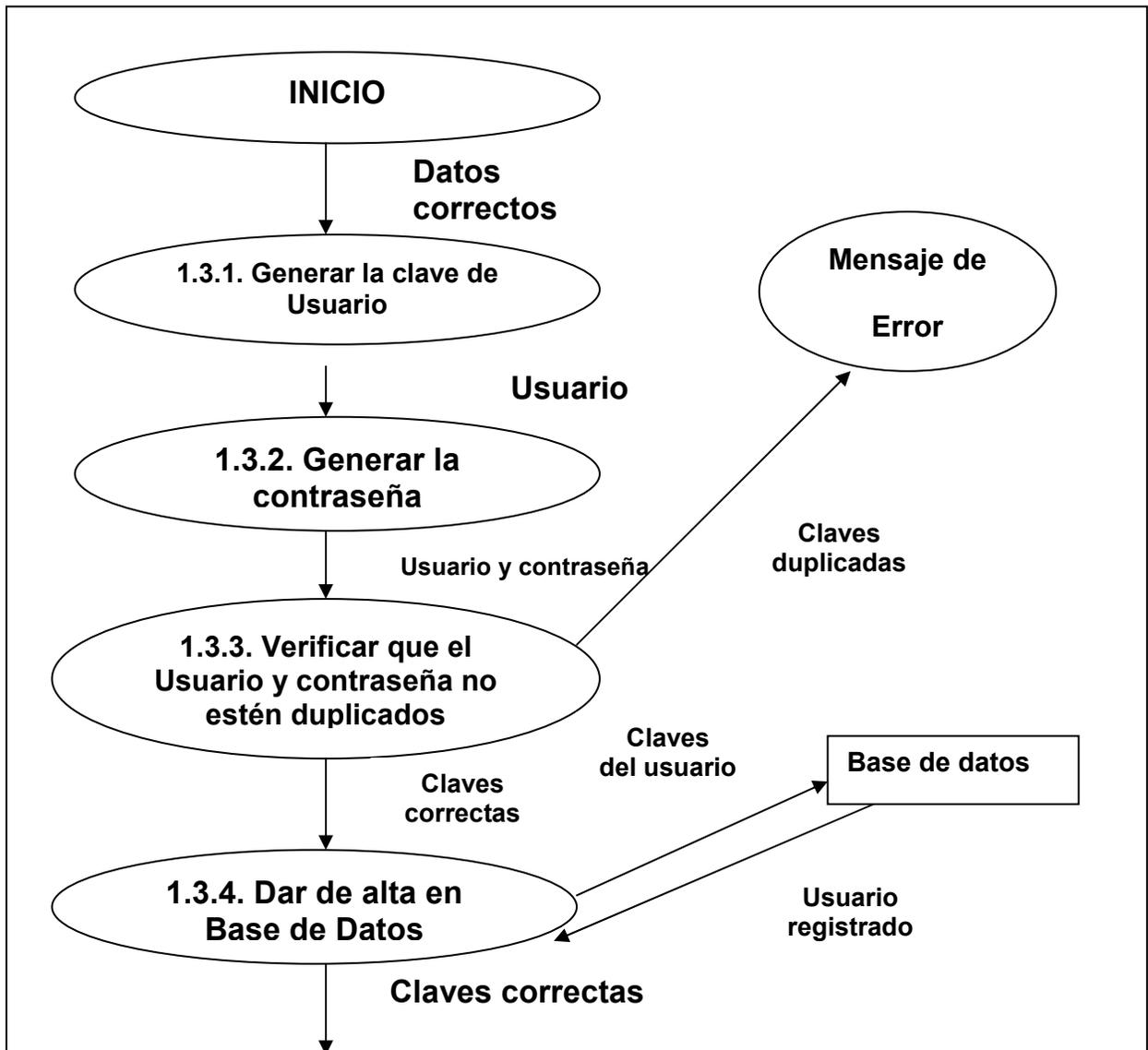


Fig. 5.3 DFD para dar de alta a un usuario en el sistema, Nivel 1.3

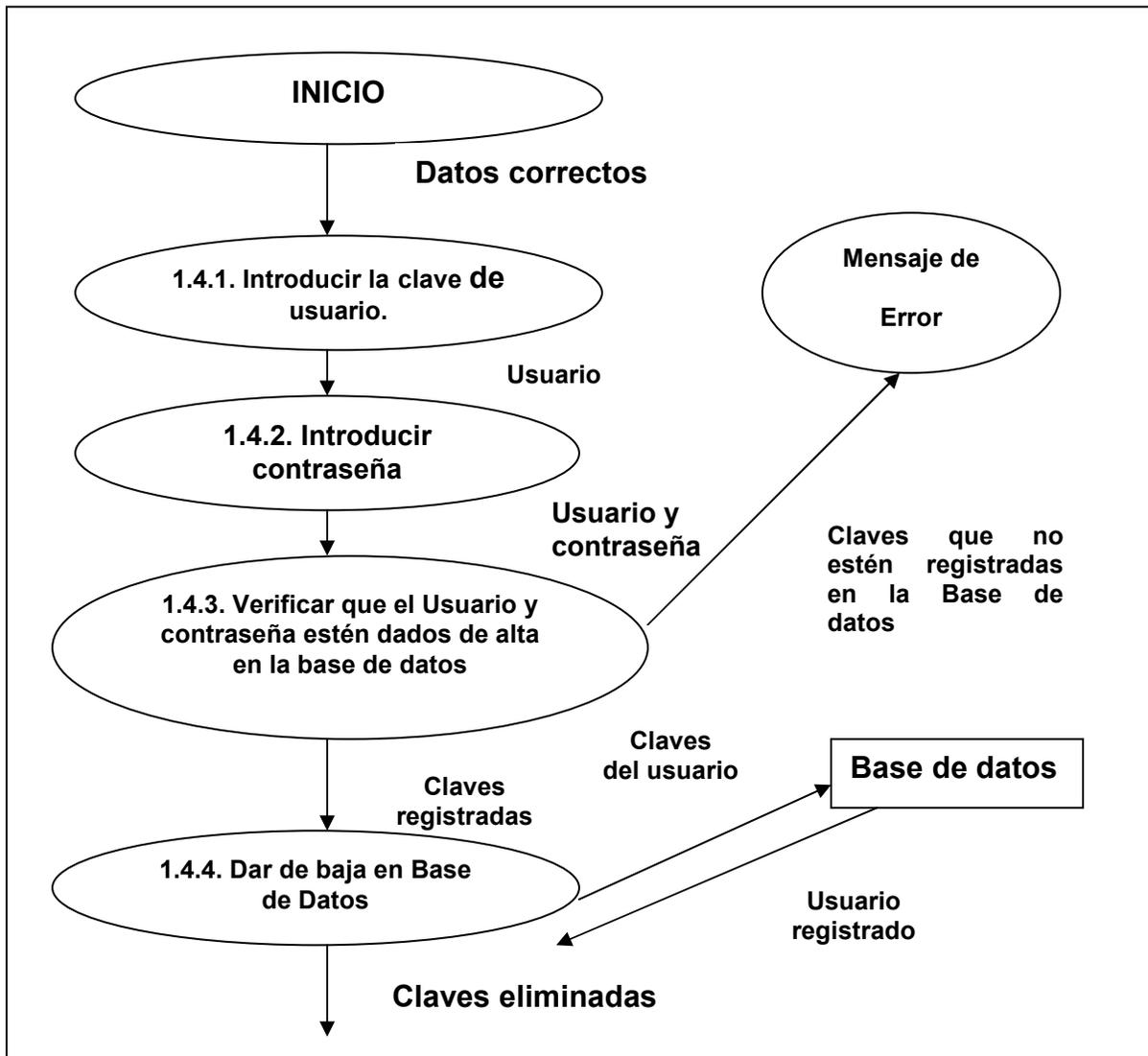


Fig. 5.4 DFD para la baja a un usuario en el sistema, Nivel 1.4

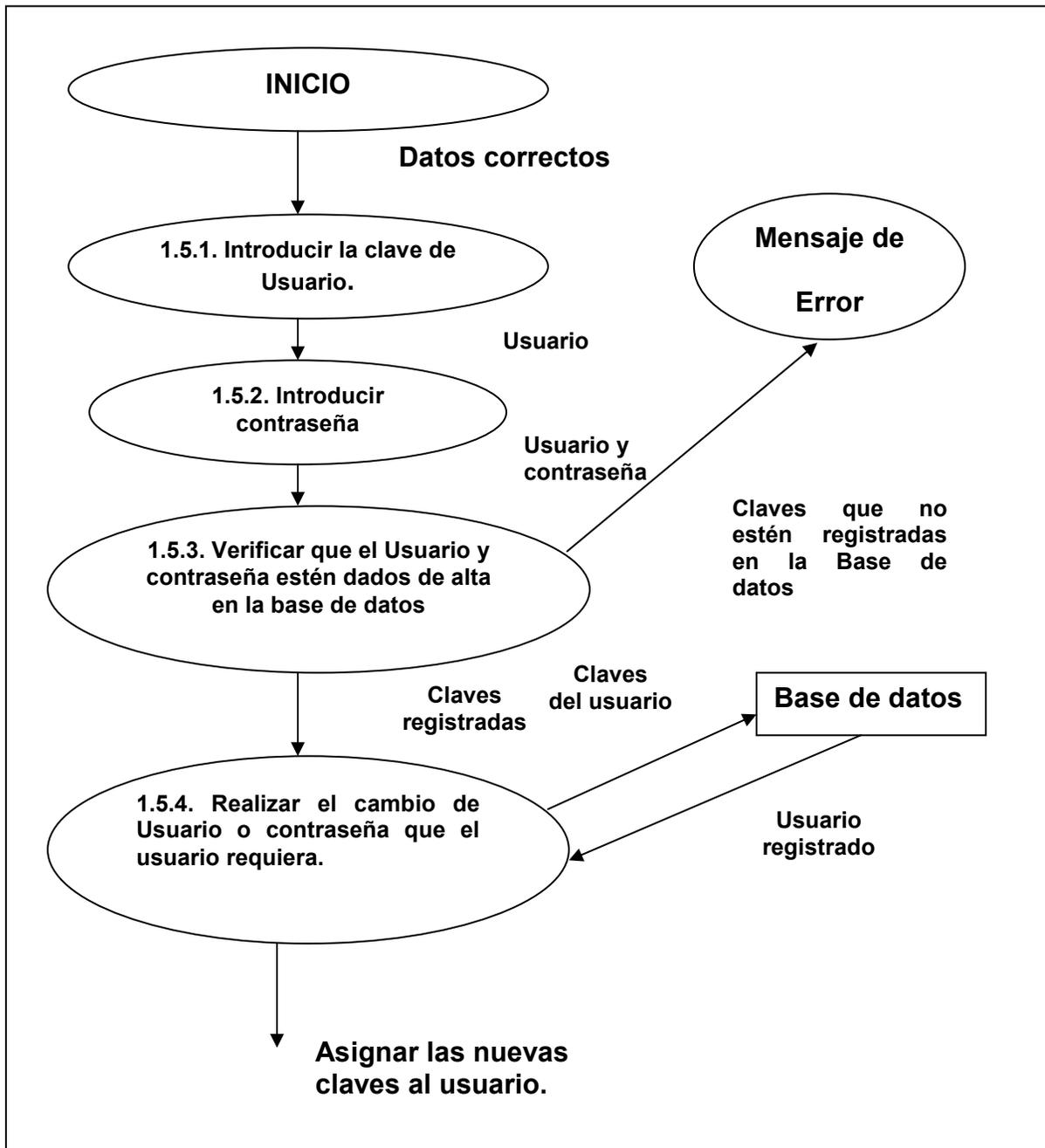


Fig. 5.5 DFD para realizar cambios a un usuario en el sistema, Nivel 1.5

Descripción de los diagramas de flujo para el mantenimiento de usuarios

En los diagramas anteriores se observa que los procesos que realiza el sistema están representados con óvalos y flechas de flujo. Estos diagramas muestran los procesos que se llevan a cabo de manera general pero que ayuda a ver claramente el orden en el que el sistema ejecuta tales procesos. A continuación se describen los procesos mostrados en los diagramas de flujo creados.

1.1. Llenar forma. Se piden los datos necesarios para poder dar de alta un usuario el cual puede ser de tipo alumno, profesor o administrador. Los datos son los siguientes: Nombre completo del usuario, número de cuenta, Ciudad, Estado, País, Teléfono y Correo Electrónico.

1.2. Validar los Datos. Se verifica que los datos introducidos por el usuario sean válidos.

1.2.1. Validar campos requeridos. Se verifica que todos los datos requeridos de la forma sean introducidos por parte del usuario. Los datos requeridos son indicados por medio del símbolo.

1.2.2. Validar el valor de los datos. Revisar que los datos introducidos en la forma sean válidos para cada uno de los campos.

1.2.3. Presentar la información a registrar. Se encarga de revisar que el(los) usuario(s) que se quiere(n) dar de alta no se encuentre(n) ya registrados en el sistema.

1.3. Dar de alta al usuario. Se dá de alta al usuario en la base de datos. Se generan las claves de acceso del usuario: un login y una contraseña. También se verifica que dichas claves de acceso no se encuentren duplicadas.

1.3.1. Generar la clave de Usuario. En el caso del alumno la clave de Usuario estará determinada por el número de cuenta que lo identifica como alumno de la ENP (Escuela Nacional Preparatoria) plantel número 7.

1.3.2. Generar la contraseña. En este caso será la contraseña que el administrador o el mismo usuario decida escoger.

1.3.3. Verificar que el usuario no esté duplicado. Con las claves de acceso ya generadas, se verifica que no se encuentre duplicado en la base de datos.

1.3.4. Dar de alta en la Base de Datos. Realiza la alta del usuario dentro de la tabla “usuarios” en la base de datos.

1.4. Enviar claves por correo. La clave de usuario y contraseña son enviados por correo electrónico a la dirección registrada por el usuario, notificando que el registro del usuario se llevó a cabo satisfactoriamente.

1.4.1. En caso de que el usuario ya no se encuentre activo en el sistema se realiza un proceso de baja en el mismo.

1.4.2. Se introduce la clave de Usuario y contraseña.

1.4.3. Se verifica que el usuario este registrado en la base de datos y tenga un estado de cancelado en el mismo, considerando la posibilidad de que el usuario nunca haya sido dado de alta en la base de datos por que se enviará un mensaje de error, ya que no es posible dar de baja lo que nunca ha sido dado de alta.

1.4.4. Después de confirmar que la clave de usuario y contraseña si están en la base de datos, se procede a eliminarlas o darlas de baja siempre y cuando así se requiera.

1.5. Considerando la posibilidad de que el usuario solicite un cambio en su clave de usuario y contraseña se incluye el proceso de cambios.

1.5.1. Se introduce la clave de usuario.

1.5.2. Se introduce la contraseña.

1.5.3. Se verifica que la clave de usuario y la contraseña estén dados de alta en la Base de Datos considerando que el usuario se encuentre activo en el sistema, en caso de no ser así el sistema mandará un mensaje de error.

1.5.4. En esta parte del DFD se realiza el cambio de la clave de usuario y contraseña de acuerdo a los requerimientos del usuario y administrador del sistema.

5.2 Creación de base de datos

Es importante definir la estructura que debe tener la base de datos utilizada en el sistema, pues de lo contrario se corre el riesgo de caer en problemas de redundancia o se albergaría información innecesaria lo que puede provocar error de datos en un futuro.

5.2.1 Tablas derivadas

Con base a las necesidades ya explicadas, se propone el conjunto de tablas con los posibles campos que se requieren para el buen funcionamiento del sistema:

asignaturas	cambiosreactivos	cambiosasignaturas
<u>clave</u> <u>version</u> descripción colegio estatus usuarioAlta fechaAlta	<u>clave</u> <u>version</u> idMateria idTema versionTema reactivo nivel opcionA opcionB opcionC opcionD opcionE justificacionaA justificacionaB justificacionaC justificacionaD justificacionaE opcionCorrecta usuarioAlta fechaAlta tipo usuarioCambio fechaCambio enviado contestado	<u>clave</u> <u>version</u> descripcion colegio estatus usuarioAlta fechaAlta tipo usuarioCambio fechaCambio enviado contestado
cambios		
<u>id</u> modulo valores tipo usuarioAlta fechaAlta		
cambioscolegios		
<u>clave</u> <u>version</u> descripcion estatus usuarioAlta fechaAlta tipo usuarioCambio fechaCambio enviado contestado		
	profesor_materias	
	<u>id</u> login clave usuarioAlta fechaAlta	
		cambiosstemas
		<u>id</u> <u>version</u> claveAsignatura tema descripcion indice usuarioAlta fechaAlta tipo usuarioCambio fechaCambio enviado contestado

cambiosusuarios
<u>login</u> <u>version</u> password nombre apellidoPaterno apellidoMaterno correo tipoUsuario estatus materias usuarioAlta fechaAlta tipo usuarioCambio fechaCambio revisado enviado contestado

datosexamen
<u>id</u> idMateria titulo idReactivos idVersionesReactivos usuarioAlta fechaAlta

temas
<u>id</u> version claveAsignatura tema descripcion indice usuarioAlta FechaAlta

reactivosexamen
<u>idExamen</u> <u>idReactivo</u> versionReactivo idVersionTemas usuarioAlta fechaAlta

colegios
<u>clave</u> <u>version</u> descripcion estatus usuarioAlta fechaAlta

correos
login correo mensaje usuarioAlta fechaAlta estatus

usuarios
<u>login</u> version password nombre apellidoPaterno apellidoMaterno correo tipoUsuario estatus materias usuarioAlta fechaAlta

profesores
<u>login</u> password nombre apellidoPaterno apellidoMaterno correo tipoUsuario estatus materias usuarioAlta fechaAlta

reactivos
<u>id</u> version idMateria idTema versionTema reactivo nivel opcionA opcionB opcionC opcionD opcionE justificacionA justificacionB justificacionC justificacionD justificacionE opcionCorrecta usuarioAlta fechaAlta

5.2.2 Diseño del Modelo Entidad-Relación

Para poder definir el diagrama de Entidad-Relación se deben considerar los criterios de derivación de tablas según las cardinalidades, son tres las reglas básicas que nos sirven para definir apropiadamente dicho diagrama:

- Todo tipo de entidad se convierte en una relación.
- Todo tipo de interrelación N a M se transforma en una relación.
- Para todo tipo de interrelación 1 a N se realiza lo que se denomina propagación de clave (regla general), o se crea una nueva relación.

Es preciso identificar entidades que interactúan con el sistema directamente, es decir, aquellos objetos que tienen atributos y pueden tener relación con alguna otra de las entidades. Por ejemplo, la entidad profesores se puede relacionar con la entidad curso mediante 1 a N, es decir, un profesor puede impartir uno o mas cursos.

Con base en lo anterior, se procede a la elaboración del Diagrama Entidad-Relación de las tablas de la base de datos utilizada, con lo que tenemos que el diagrama resultante es:

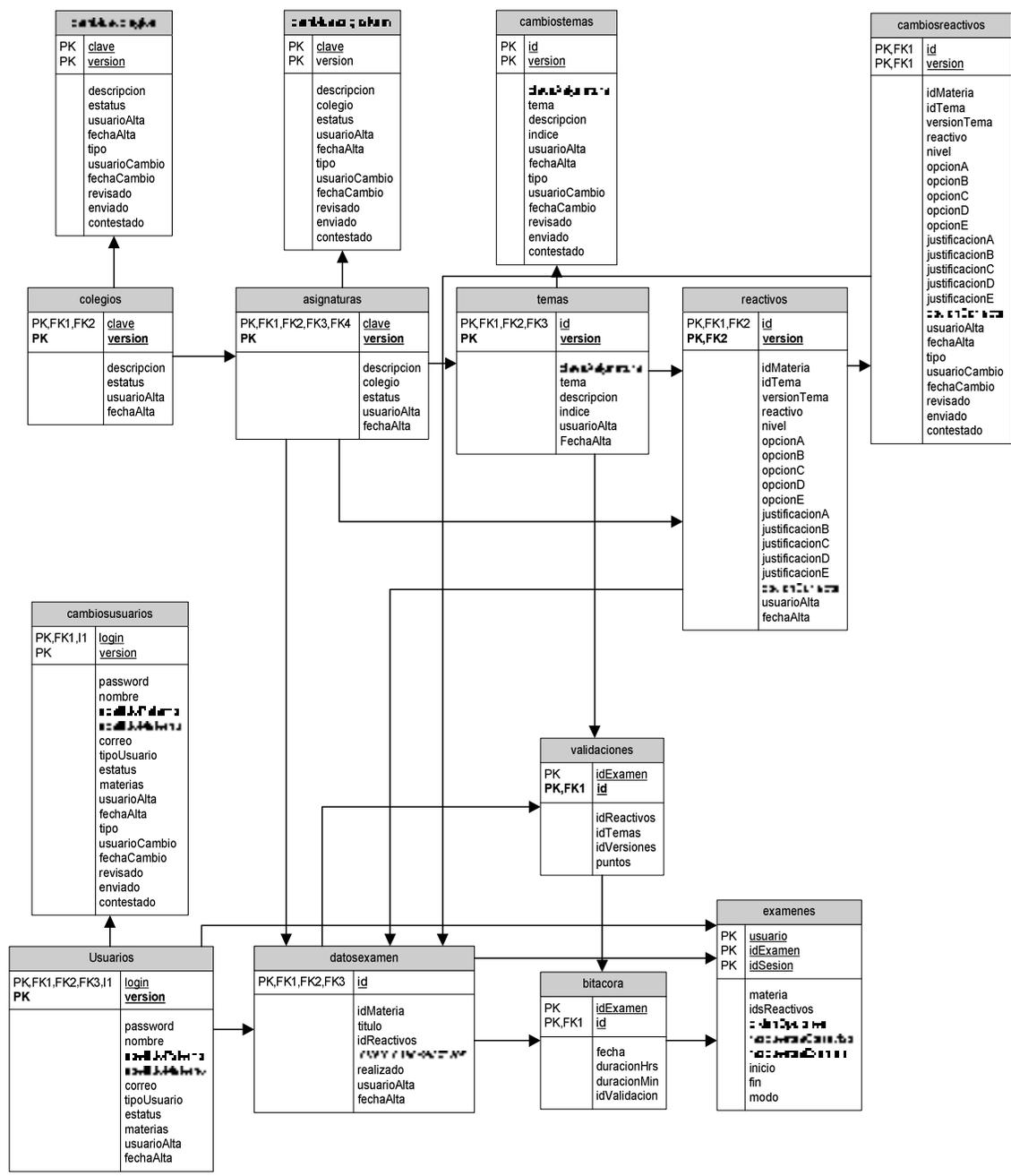


Fig. 5.6 Diagrama de entidad-relación

5.3 Diseño de la arquitectura de los módulos del sistema

La arquitectura del sistema y el cómo quedarán acomodadas todas las partes dentro del mismo debe ser clara. Los usuarios se comunican con el sistema a través de un navegador Web, y éste a su vez, ejecuta todas las instrucciones JavaScript que tienen el propósito fundamental de validar y dar “dinamismo” a las páginas del lado del usuario final del sistema. Además, dicho navegador se comunica directamente con el servidor Web el cual es el responsable de manejar las peticiones y respuestas entre la máquina del usuario y la máquina donde radica el sistema (servidor), es decir, manipular los flujos de datos que dicha comunicación conlleve. Es importante hacer notar que el acceso a la base de datos se logra a través de código PHP. Como se puede observar en la Fig. 5.6 los usuarios finales (administrador, alumno y profesor) nunca interactúan directamente con la base de datos o la programación en PHP.

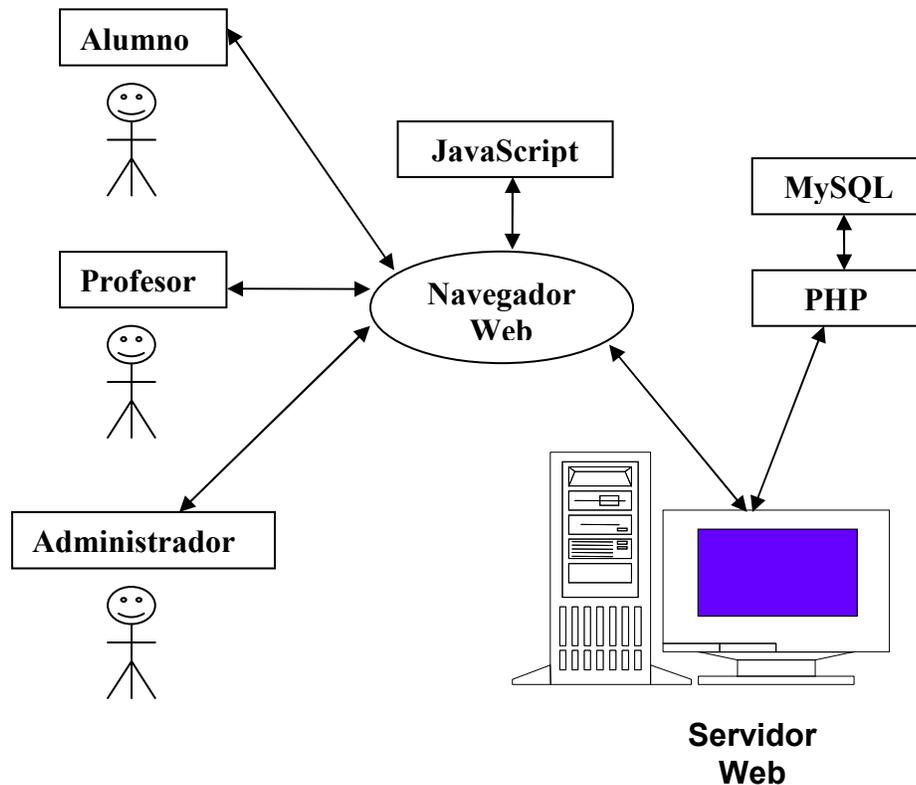


Fig. 5.7 Arquitectura del sistema

El sistema fue desarrollado bajo un patrón de diseño MVC (Modelo-Vista _ Controlador) por lo cual, la única parte que interactúa directamente con el usuario es la “vista” o interfaz, para esta situación todas las vistas están representadas mediante scripts, los cuales, para cumplir su función de “vista” no pueden generar ningún tipo de información por sí solos, es decir, los datos que ingrese cierto usuario en una interfaz serán mandados a un controlador que definirá el modelo (función, método) donde se procesara la petición, una vez procesada el modelo regresa sus resultados a la página solicitada el cual etiquetara los resultados de tal manera que la siguiente “vista” que se invoque, el usuario pueda comprender y entender la respuesta a su petición.

Diseño utilizado para la generación de exámenes

Una vez que el alumno entre al sistema este debe ser registrado en una tabla de usuarios que estén conectados a la base de datos para evitar que inicie otra sesión, luego, éste debe elegir el examen a realizar, solo así el examen será generado y presentado al usuario para la aplicación de una manera transparente para quien lo realiza.

Para la generación de exámenes en esta aplicación se necesita una tabla dentro de la base de datos que almacene el id de cada pregunta que esté en uso y así se asegurará que los exámenes generados simultáneamente no sean iguales debido a que no podrían tomar las preguntas que tengan el id registrado en dicha tabla, si en la base de datos haya suficientes preguntas disponibles para generar un examen de cierto curso. Si lo anterior no es posible, la generación de exámenes consistirá primero en mostrar el orden de las preguntas de forma aleatoria, de la misma manera para las respuestas asociadas a cada pregunta.

Capítulo 6

Programación y pruebas del sistema

6.1 Estudio del software utilizado para el desarrollo: EasyPHP

6.1.1 ¿Qué es EasyPHP?

Es un paquete que instala todos los componentes necesarios para trabajar con páginas PHP, y en nuestro caso corresponden a las herramientas que antes indicamos se utilizarían para la creación del sistema, la versión que se manejó fue EasyPHP 1.8, ésta versión instala los siguientes componentes:

- Apache 1.3.33
- PHP 4.3.10
- phpMyAdmin 2.6.1
- MySQL 4.1.9

La descarga de EasyPHP se encuentra disponible en la siguiente dirección:
<http://www.easyph.org>.

Tenemos que aclarar que EasyPHP es un software que trabaja en el sistema operativo Windows. Se decidió utilizar esta herramienta, a pesar de que la idea es hospedar el sistema en una plataforma Linux, porque las computaras que estaban disponibles para realizar el desarrollo tenían Windows instalado. Sin embargo, lo anterior no afecta de ninguna manera a la versión final del sistema debido a que todo el software requerido para el funcionamiento del sistema tiene su respectiva versión para Linux, es decir, aprovechamos la propiedad de portabilidad que nos dan tales herramientas de desarrollo.

En la siguiente figura se ilustra la interfaz de la versión utilizada de EasyPHP.



Fig. 6.1 Interfaz de EasyPHP con el menú de MySQL desplegado

6.1.1.1 MySQL 4.1.9

A pesar de la potencia de MySQL, éste tiene un inconveniente: su interfaz de trabajo es en modo consola, es decir, a línea de comando. Esto resulta en muchas ocasiones un procedimiento engorroso por lo que se decidió trabajar con phpMyAdmin, que presenta una interfaz mucho más amigable. El aspecto de dicha interfaz es la que se muestra en la siguiente figura:

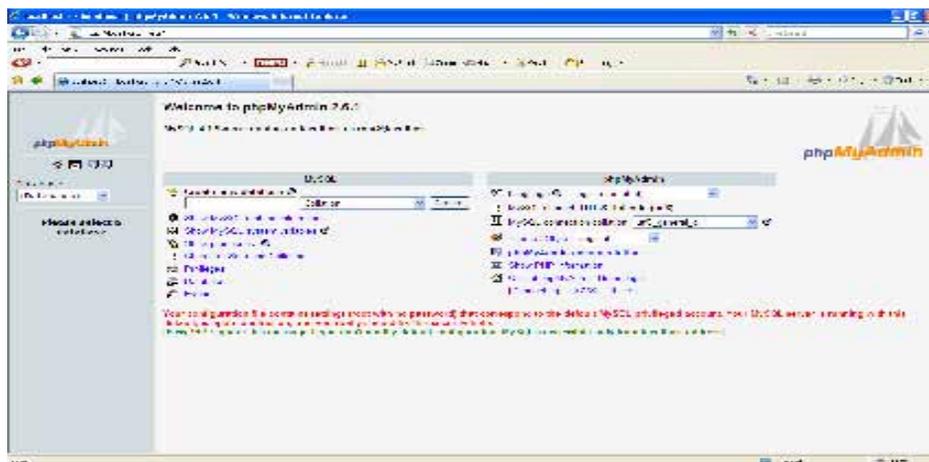


Fig. 6.2 Interfaz de phpMyAdmin

6.1.1.2 Servidor Apache 1.3.33.

El servidor Web Apache es uno de los usados. Este servidor trabaja con lenguajes como CGI, Perl, PHP y otros lenguajes script. Existen versiones tanto para sistemas operativos Unix (Linux, Solaris, Mac-OS), así como con Windows.

Las principales características que tiene este servidor Web son las siguientes:

- Se puede utilizar tanto en sistemas Unix como en sistemas Windows, existiendo versiones diferentes para cada uno de los sistemas.
- Es de código abierto y gratuito.
- Toda la configuración se basa en su archivo *httpd.conf*. Este archivo es editable con cualquier editor de textos.
- Servidor proxy integrado.
- Soporte de SSL (módulo OpenSSL y *mod_ssl*).

6.2 Manual de usuario del sistema

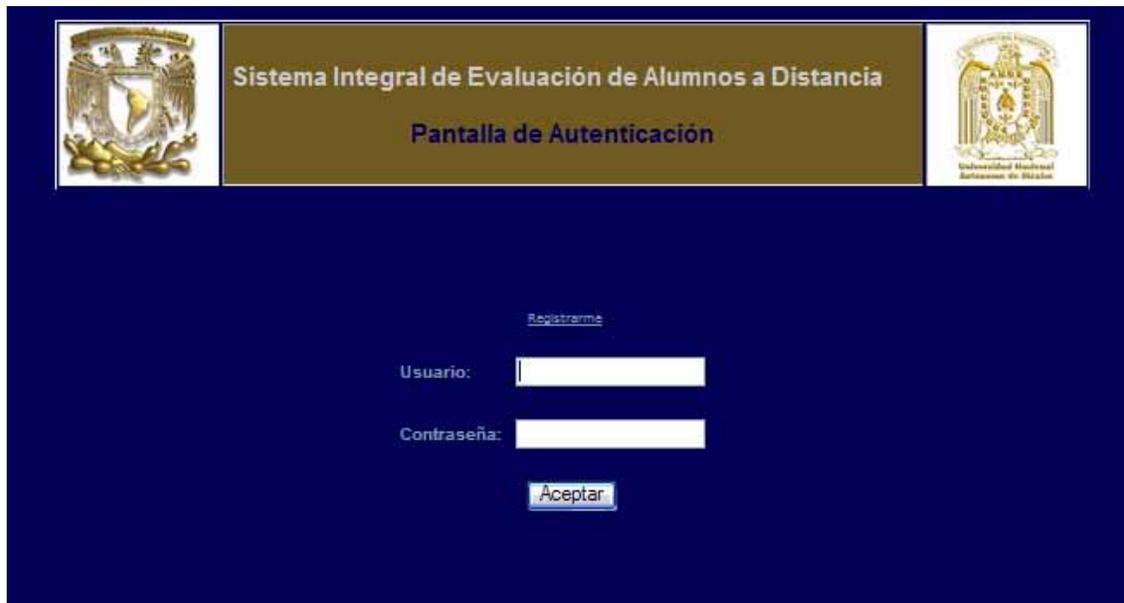
6.2.1 Tipos de sesión

El usuario entra al sistema mediante un navegador Web ingresando el siguiente URL:

http://{dominio}/siead/

en donde {dominio} es el nombre que identifica a un sitio web, en éste caso, se proporcionará el nombre de tal dominio una vez que lo tengamos asignado.

La primera interfaz que se muestra contiene un formulario en donde el usuario ingresa su login y contraseña, para establecer y/o iniciar una sesión, tal y como se observa en la siguiente pantalla:



The screenshot shows a dark blue background with a gold header bar. The header bar contains the text "Sistema Integral de Evaluación de Alumnos a Distancia" and "Pantalla de Autenticación" in white. On either side of the header bar is a gold crest. Below the header bar, there is a "Regístrate" link in blue. Below the link are two white input fields: "Usuario:" and "Contraseña:". Below the input fields is a blue "Aceptar" button.

Fig. 6.3 Pantalla de autenticación

En caso de que el usuario y/o contraseña no sean válidos, el sistema manda un mensaje de error como se muestra en la siguiente ventana.

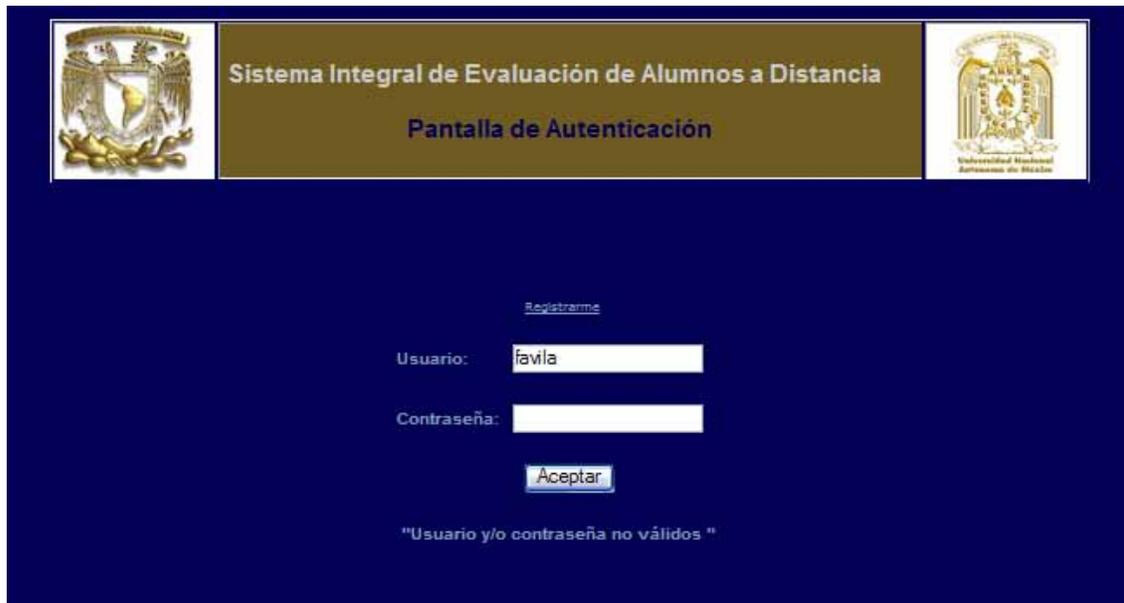


Fig. 6.4 Mensaje para un usuario no válido

Si el usuario no puede iniciar sesión en el sistema por no estar dado de alta en la base de datos, tendrá la opción de registrarse dando clic en la liga *Regístrarme* que se encuentra en la Pantalla de Autenticación, lo que llevará al usuario a la ventana de Registro para examen (ver la figura de abajo). Sin embargo, cabe señalar que el registro del usuario a través de tal opción, le asignará automáticamente los privilegios mínimos para acceder al sistema.



The screenshot shows a web interface for user registration. At the top, there is a header with the text 'Sistema Integral de Evaluación de Alumnos a Distancia' and 'Registro de usuarios'. The header is flanked by two logos of the 'Universidad Nacional Autónoma de México'. Below the header, there is a form with the following fields and buttons:

- Usuario:** A text input field followed by a 'Buscar' button.
- Nombre (s):** A text input field.
- Apellido paterno:** A text input field.
- Apellido materno:** A text input field.
- Correo(s):** A text input field.
- Contraseña:** A text input field.
- Confirmar Contr.:** A text input field.
- Guardar:** A button located to the right of the 'Nombre (s)' field.
- Salir:** A button located below the 'Guardar' button.

Fig. 6.5 Formato para registrarse como usuario del sistema

En esta pantalla, puede ingresar su número de cuenta si es un alumno, o bien, su número de empleado si es trabajador o profesor de la institución. La opción de *Buscar* le permite al usuario verificar que el número de usuario que está ingresando está disponible, o bien, si existe registro del mismo.

La opción de *Guardar* permite guardar los datos en la base de datos, si es que se ha cumplido con la validación que el sistema realiza de la información ingresada; en su defecto, se le enviarán los mensajes necesarios para indicar que no se han ingresado los datos solicitados, o bien, si hay valores incorrectos en los campos del formulario.

La opción de *Salir* nos regresa a la pantalla de autenticación.

Nota: debe considerarse que el sistema realiza la validación de los datos en cada pantalla del lado del usuario (con scripts elaborados en JavaScript) con la finalidad de evitar que el usuario envíe la información al servidor y que al tratar

de almacenarlos, se detecte que no son correctos o faltan datos. Con esto se ahorra tiempo, pues al no poderse realizar el proceso de almacenar los datos, el proceso de captura tendría que repetirse, para nuevamente enviar los datos. Una vez mencionado lo anterior, en lo que resta de esta guía, y con fines de no hacer complicado el contenido de la misma, se entenderá que los datos de cada pantalla serán validados por el sistema, y éste enviará los mensajes oportunos según sea el caso.

El manejo de sesiones es importante en este sistema para poder mantener la consistencia de los datos en cada petición del usuario a través de los scripts debido a que, por sí solo, el protocolo HTTP no tiene forma de diferenciar las peticiones de cada usuario que estén usando el sistema.

Cuando el alumno ingresa sus datos en la página de autenticación y envía los datos, se hace un llamado a la página que valida los datos proporcionados por el usuario consultando la tabla “usuarios” (dicha tabla contiene los campos usuario y contraseña de todos los usuarios registrados que pueden acceder al sistema). Si los datos son válidos, verifica que dicho alumno esté activo (es decir, puede estar registrado, pero no activo) de lo contrario no permite la entrada a la interfaz que muestra el menú principal donde se le presentarán las opciones a las que puede acceder de acuerdo al perfil de usuario que tenga. El tipo de usuario puede ser de categoría 1, 2 ó 3 correspondiente al usuario alumno y profesor respectivamente. Para un usuario de tipo alumno, las opciones a las que tiene acceso se muestran en la siguiente figura.

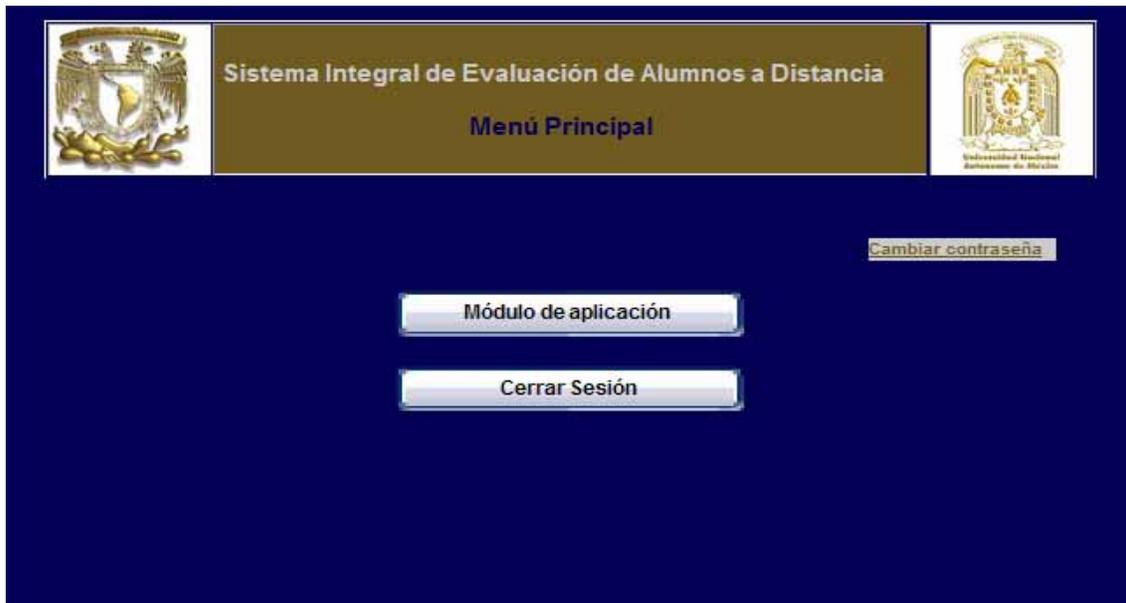


Fig. 6.6 Opciones disponibles para un usuario de tipo alumno

Para un usuario de tipo profesor, las opciones habilitadas se muestran la siguiente interfaz:

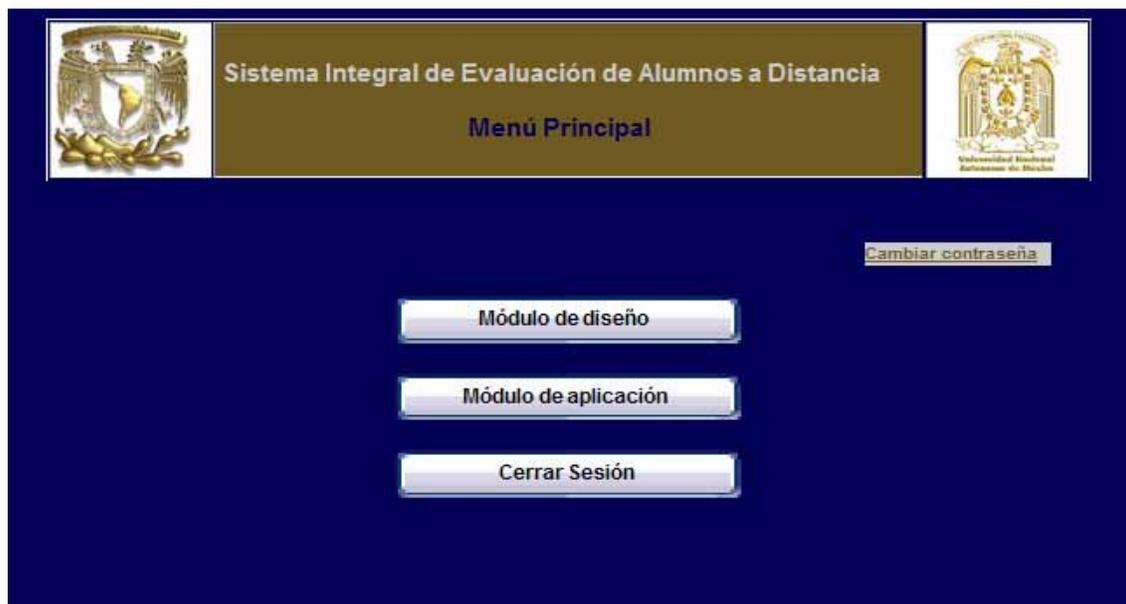


Fig. 6.7 Opciones disponibles para un usuario de tipo profesor

Finalmente, al usuario de tipo administrador, se le da acceso a todos los recursos del sistema como se observa a continuación:

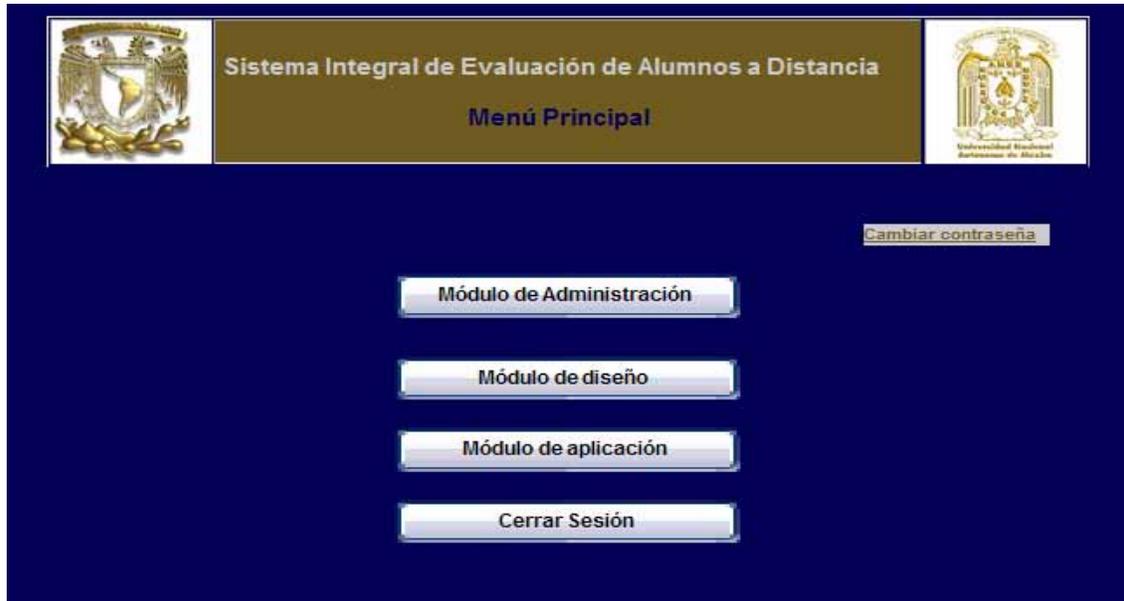


Fig. 6.8 Opciones disponibles para un usuario de tipo administrador

6.2.2 Módulo de administración

El módulo de administración se divide en las siguientes secciones: *Colegios*, *Materias*, *Reactivos*, *Usuarios* y, finalmente, *Cambios*; como se muestra en la siguiente imagen:

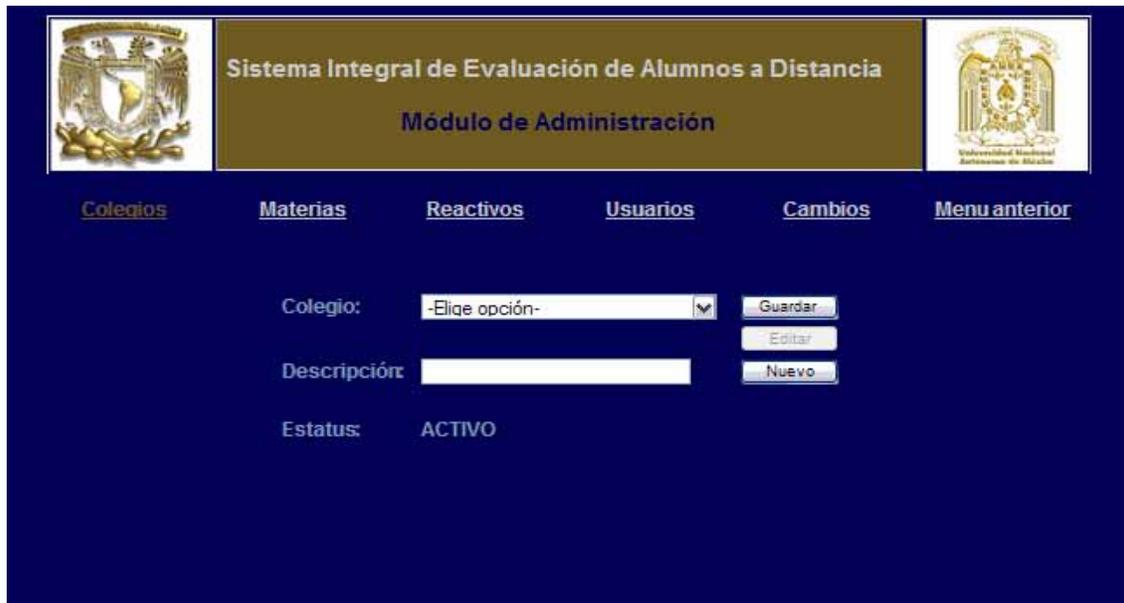


Fig. 6.9 Opciones del módulo de administración

El módulo de administración inicia con la sección de *Colegios*. Aquí es en donde podemos agregar un nuevo colegio, o bien, actualizar los datos de uno existente.

El procedimiento para dar de alta un colegio es el siguiente: si no se ha seleccionado algún colegio en la lista de *Colegio*, basta con indicar la descripción del nuevo colegio y seleccionar la opción *Guardar*. Si se está mostrando la información de un colegio existente, tendrá que usar el botón *Nuevo*, el cual limpiará los campos del formulario para dar poder capturar la descripción del nuevo colegio. Las condiciones para agregar un nuevo colegio son las que se ven en la figura de arriba.

Nota: como puede notarse en la imagen arriba mostrada, cuando se está en modo de alta de un registro, la opción *Editar* está deshabilitada, lo que no sucede con el botón *Nuevo* cuando se tienen cargados los datos de un registro, sin embargo, si han modificado tales datos y se quiere agregar un nuevo

registro, se dará aviso al usuario que no se han guardado los cambios y él decidirá si desea descartar los cambios o cancelar la acción de alta y guardar primero los cambios. Este procedimiento es constante en todas aquellas interfaces en donde se tengan las opciones de *Editar* y *Nuevo* para los registros según corresponda. Ahora bien, la opción *Guardar* si está deshabilitada en modo consulta, este modo es en el que se inicia la carga de un registro o cuando se está editando y se guardan los cambios, o cuando se guarda un nuevo registro, automáticamente se muestra el registro en modo consulta, con lo que se indica que la actualización o alta del registro se ha realizado.

Ahora bien, si se desea consultar los datos de un colegio, se debe seleccionar de la lista *Colegio* para cargar los datos relacionados al mismo. La siguiente pantalla muestra como se presentarían los datos en modo consulta.

The screenshot shows a web interface for the 'Sistema Integral de Evaluación de Alumnos a Distancia' (SIEAD) in the 'Modulo de Administración' (Administration Module). The interface is dark blue with gold accents. At the top, there are two logos of the 'Universidad Nacional Autónoma de México' (UNAM) flanking the title. Below the title is a navigation menu with the following items: Colegios, Materias, Reactivos, Usuarios, Cambios, and Menu anterior. The main content area displays a form for a 'Colegio' (School). The 'Colegio' field is a dropdown menu currently showing 'INFORMÁTICA'. To its right are three buttons: 'Guardar' (disabled), 'Editar', and 'Nuevo'. Below this, the 'Descripción' (Description) field shows 'INFORMÁTICA'. At the bottom, the 'Estatus' (Status) field shows 'ACTIVO'.

Fig. 6.10 Modo consulta de la opción colegios

En el modo de edición o actualización, se puede modificar la descripción y/o estatus del colegio; se tendría algo como la siguiente interfaz:

The screenshot shows the 'Módulo de Administración' interface for updating 'Colegios'. At the top, there are two logos and the title 'Sistema Integral de Evaluación de Alumnos a Distancia'. Below the title is a navigation menu with links: 'Colegios', 'Materias', 'Reactivos', 'Usuarios', 'Cambios', and 'Menu anterior'. The main form contains three rows of input fields: 'Colegio:' with a dropdown menu showing 'INFORMÁTICA', 'Descripción:' with a text input field containing 'INFORMÁTICA', and 'Estatus:' with a dropdown menu showing 'ACTIVO'. To the right of these fields are three buttons: 'Guardar', 'Editar', and 'Nuevo'.

Fig. 6.11 Modo actualización de la opción colegios

La sección de *Materias* tiene el siguiente formato:

The screenshot shows the 'Materias' interface. At the top, there are two logos and the title 'Sistema Integral de Evaluación de Alumnos a Distancia'. Below the title is a navigation menu with links: 'Colegios', 'Materias', 'Reactivos', 'Usuarios', 'Cambios', and 'Menu anterior'. The main form contains several sections: 'Clave:' with a text input field and a 'Buscar' button; 'Estatus:' with a dropdown menu showing 'ACTIVO'; 'Materia:' with a text input field and three buttons: 'Guardar', 'Editar', and 'Nuevo'; 'Colegio:' with a dropdown menu showing '-Elige opción-'; 'Tema:' with a text input field and two buttons: 'Nivel' and 'Subnivel'; and 'Cambios:' with a text input field and two buttons: 'Habilitar' and 'Eliminar'.

Fig. 6.12 Interfaz de la opción materias

Los botones *Guardar*, *Editar* y *Nuevo* realizan las funciones que describen sus nombres para los datos relacionados a una materia, los cuales son : clave, descripción, colegio, temas y subtemas (temario).

El proceso para registrar una nueva materia es : sino se ha cargado el registro de una materia existente, basta con llenar los campos *Clave*, *Materia* y *Colegio* con la clave y descripción de la materia y el colegio al que pertenece. Los campos *Tema* y *Cambios* sirven para agregar y modificar respectivamente los temas y/o subtemas que componen el temario de la materia.

Nota: aunque no es obligatorio crear el temario de la materia para poder guardar el registro, resulta evidente la necesidad de hacerlo debido a que los reactivos siempre se asignarán a un tema, es decir, un reactivo depende de un tema, y sino hay tema alguno, tampoco se podrán registrar reactivos de la materia en cuestión. Aún más, los temas de los que dependan los reactivos, no pueden ser del primer nivel del temario, es decir, de las unidades; esto es para que los reactivos no dependan de temas muy generales.

Para crear el temario basta con capturar el texto del tema que se desea en el campo *Tema* y dar clic en el botón *Nivel* o *Subnivel* para agregarlo al temario como tema o subtema respectivamente. Una cosa que hay que considerar, es que para el caso de querer agregar un tema como subtema de algún otro, debe de estar seleccionado el tema del que dependerá; la forma de seleccionar un tema es haciendo clic sobre el texto del mismo, se podrá notar cuando un tema está seleccionado porque aparece una flecha al lado izquierdo de él. Observe el siguiente ejemplo, en donde se crea un tema y después un subtema.

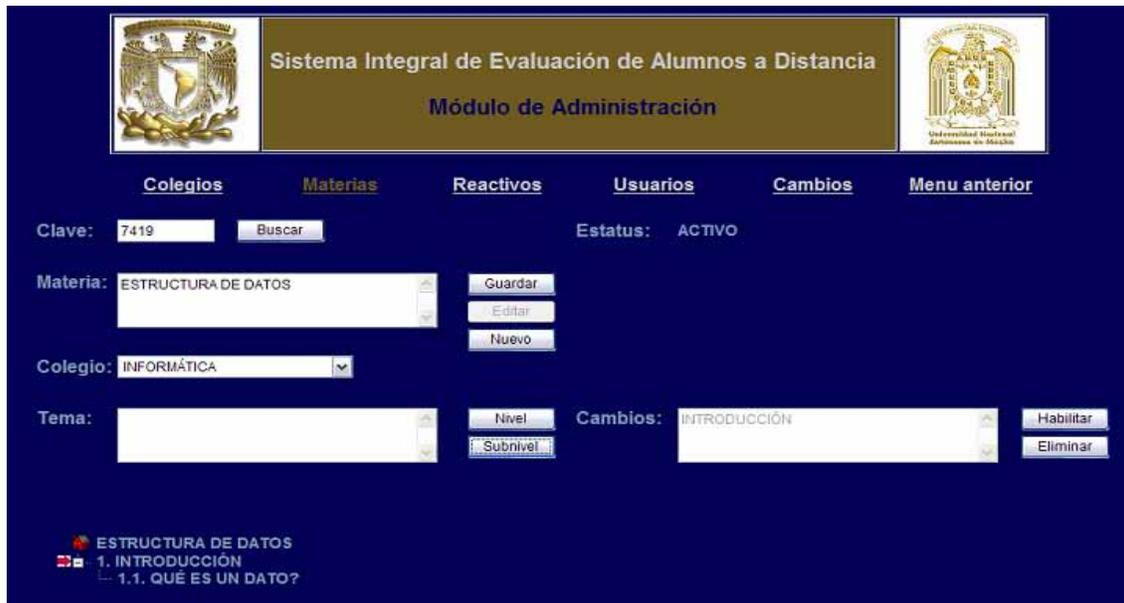


Fig. 6.13 Creación de un temario para una asignatura

De esta pantalla podemos comentar lo siguiente: observe que los temas están numerados, sin embargo, en realidad sólo se ingresó el texto de los temas, la numeración la asigna el sistema conforme se agregan los temas, e incluso, se actualizan cuando es necesario como por ejemplo, cuando se agrega o elimina un tema en medio de otros, obviamente, los temas que aparecen debajo del tema involucrado, deben actualizarse en su numeración.

Otra cosa que se puede comentar sobre la pantalla anterior, es hacer referencia a que el tema que está seleccionado aparece (aunque en forma no editable) en el campo *Cambios*; esto sirve para dos cosas, para dar la forma de modificar el tema o bien, eliminarlo. Para el primer caso, es suficiente presionar el botón *Habilitar* y entonces se podrá cambiar el texto del tema; los cambios que se hagan se verán reflejados inmediatamente, como se muestra en la siguiente imagen. Para eliminar el tema seleccionado, simplemente se debe presionar el botón de *Eliminar* y confirmar que se desea eliminar el tema.

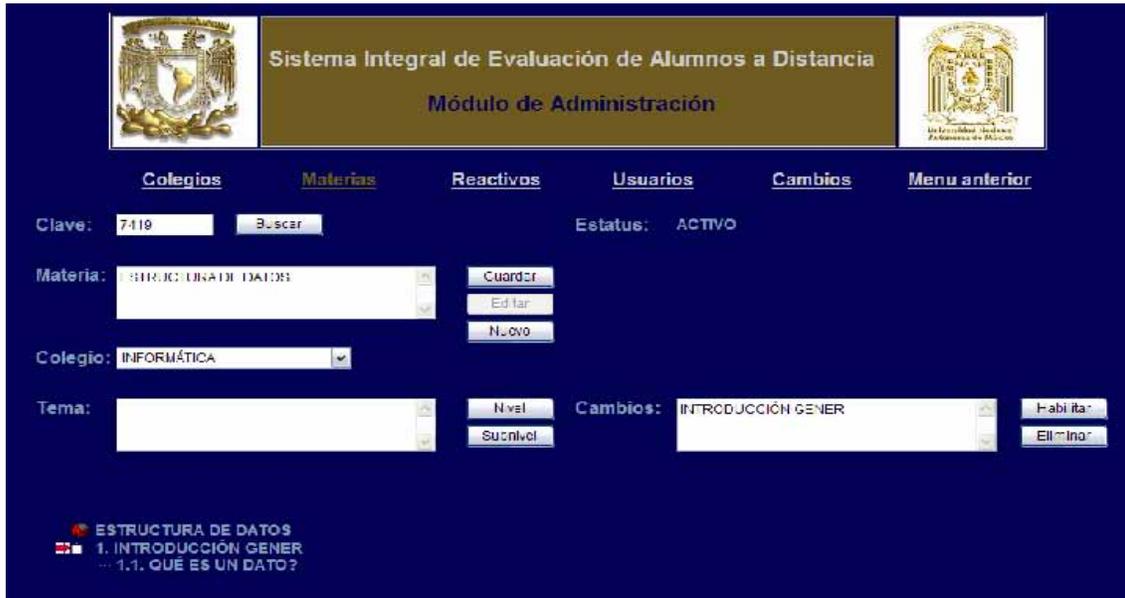


Fig. 6.14 Actualización de un temario

También, con respecto a la creación del temario, puede notarse que al agregar un subtema a un tema, se crea un botón para contraer o expandir la visualización de los subtemas, es decir, si el botón que se muestra al lado izquierdo del tema del que depende(n) otro(s) tema(s) contiene un signo de menos (-), eso significa que si se hace clic en el botón, los subtemas no se mostrarán, o bien, si el botón muestra un signo de más (+), ello quiere decir que al hacer clic sobre él, se visualizarán los temas que dependen del tema.

Para finalizar la sección de *Materias*, describiremos la función del botón *Buscar*, el cual abre una ventana nueva que lista todas las materias y que nos permite hacer un filtro por materias que pertenezcan a un colegio específico y/o por nombre de la materia. Para este último caso, al ir escribiendo el nombre de la materia, la lista se reducirá dejando en la lista las materias que contengan el texto sugerido. Lo anterior se ilustra en la siguiente gráfica.



Fig. 6.15 Búsqueda de asignaturas

El botón *Asignar*, cuando se ha seleccionado una materia de la lista, carga los datos en la pantalla de *Materia* y cierra la ventana de búsqueda. El botón *Nuevo* limpia los campos de la pantalla de *Materia* para iniciar la captura de un nuevo registro y cierra la ventana de búsqueda. El botón *Cerrar* simplemente cierra la ventana de búsqueda.

La sección de *Reactivos* es como se muestra en seguida:

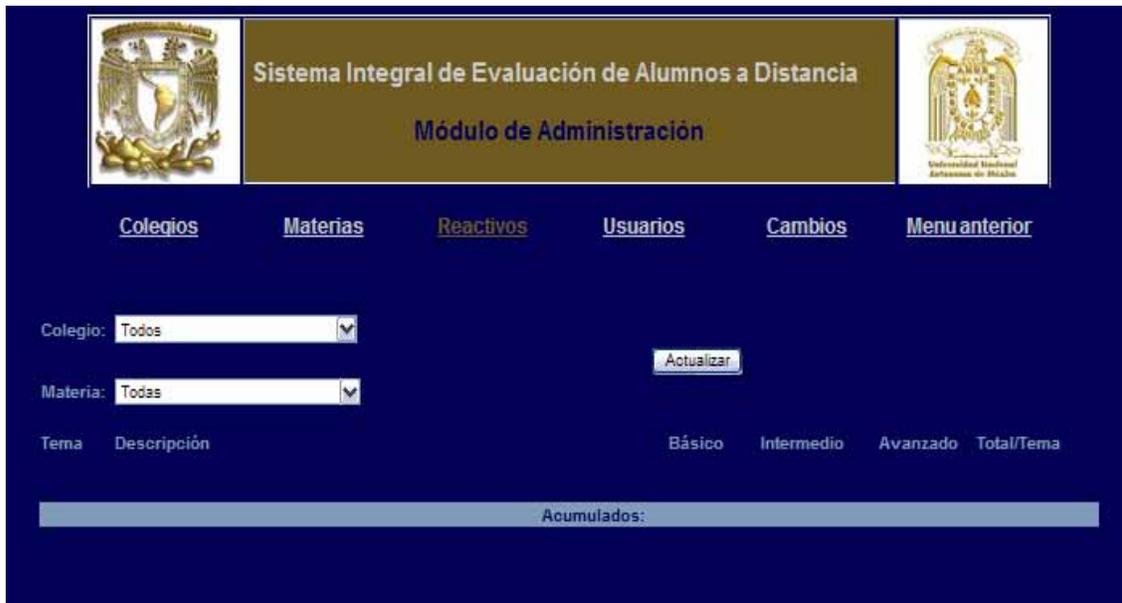


Fig. 6.16 Pantalla para la sección de reactivos

Esta pantalla muestra un resumen de los reactivos que contiene una materia ; para ver dicho resumen se debe seleccionar la materia deseada en el campo *Materia*, campo que inicialmente contiene todas las materias de todos los colegios. El campo *Colegio* contiene los colegios registrados, y al seleccionar un colegio de la lista, el campo materia sólo mostrará las materias que pertenecen a tal colegio. Una vez seleccionada una materia, se muestra el resumen de los reactivos que tiene relacionados. Observe la siguiente figura.

Tema	Descripción	Básico	Intermedio	Avanzado	Total/Tema
1.	TEMA1	1	3	2	6
1.1.	TEMA1.1	0	1	1	2
1.2.	TEMA1.2	1	2	1	4
2.	TEMA2	1	0	0	1
2.1.	TEMA 2.1	1	0	0	1
3.	TEMA3	1	3	3	7
3.1.	TEMA 3.1	1	3	3	7
4.	TEMA4	0	0	0	0
Acumulados:		3	6	5	14

Fig. 6.17 Resumen de reactivos de una asignatura

Como puede verse, el resumen se realiza con base en el temario de la materia. Las líneas cafés indican las unidades. Los reactivos se clasifican en 3 categorías: básico, intermedio y avanzado. El resumen se hace por tema, por unidad y por temario (éste último es el que está resaltado con la franja azul) considerando la clasificación de los reactivos; además de la columna *Total/Tema*, que nos da la suma total de los reactivos de todos los niveles según corresponda al contexto (tema, unidad o temario).

El botón *Actualizar* carga nuevamente la información de los reactivos relacionada a la materia que se está mostrando, esto es de utilidad cuando se han agregado, eliminado, o modificado en su clasificación algún(os) reactivo(s), lo que modificaría el resumen que había visto el usuario en un principio.

Al hacer clic en cualquier número del resumen, se abre una ventana la cual muestra la lista de reactivos que pertenecen al(los) tema(s) que correspondan al nivel del resumen que se solicita. Para ejemplificar lo siguiente, considere

que se desea ver la lista de todos los reactivos de la unidad 1 de la materia mostrada en la figura anterior; entonces, debemos hacer clic en el número 6 que está en la columna *Total/Tema* del tema 1, sombreado con línea café por ser un tema de primer nivel (“unidad”), con lo que veremos la siguiente lista.



Fig. 6.18 Búsqueda de reactivos

Explicaremos los elementos de la pantalla de *Búsqueda de reactivos*. Un dato es el número de reactivos que se listan. Los botones *Nuevo* y *Cerrar* abren una nueva ventana para capturar un nuevo reactivo y cierran la ventana actual respectivamente. Además, se muestran tres datos de los reactivos, el tema al que pertenecen, su descripción (cuando no cabe en el espacio reservado para mostrarla, se ponen tres puntos suspensivos para indicar tal circunstancia), y nivel. Estos datos también son criterios de búsqueda, es decir, se puede indicar en los campos de tales columnas los valores que se desean cumplan los reactivos que se mostrarán. De hecho, nosotros al acceder desde el resumen de los reactivos a la lista de los reactivos, implícitamente hemos indicado un criterio de búsqueda, en el ejemplo de la pantalla de arriba, se mostraron los

La *Pantalla de reactivo* contiene todos los datos que pertenecen al reactivo así como los botones para guardar, editar, eliminar y crear un nuevo reactivo, además del botón para cerrar la ventana de *Pantalla de reactivo*.

Considerando que los datos obvios no requieren explicarse, sólo lo haremos con los siguientes: *Fuente*, se refiere a la tabla en la que está almacenado el reactivo, en ésta sección todos serán reactivos *Vigentes*, es decir, todos serán la última versión de los reactivo. La versión de un reactivo es consecutiva según el número de cambios que ha sufrido desde su creación, la versión 1 es la que se asigna al reactivo cuando se crea. Cuando un reactivo se modifica o elimina, todos los datos que tenía en la versión anterior a los cambios o al momento de haber sido eliminado, se almacenan en otra tabla que es la fuente *Cambios*. Más adelante se aclarará la necesidad de este funcionamiento.

El campo *Id-Versión* nos informa sobre el id y versión del reactivo, el id es la clave del reactivo, es un número único y consecutivo conforme se almacenan los reactivos en la base de datos, es asignado automáticamente por el sistema.

Nota: el autor y fecha de alta de un reactivo es asignado automáticamente con base a la sesión del usuario que está usando el sistema y a la fecha actual. Es importante mencionar que, cuando un reactivo es modificado, en la versión que tiene los cambios del reactivo, el autor será el usuario que lo ha modificado, y la fecha de alta, la fecha en que se modificó. Esto resulta relevante porque los usuarios que no tienen privilegios de administrador, sólo pueden modificar los reactivos que tienen como autor a dicho usuario.

Para finalizar la explicación de la *Pantalla de reactivo* mencionaremos que, los campos de la columna *Justificación*, son para ingresar la justificación del

porqué el reactivo es o no es correcto; contemplando que en algún momento se necesite realizar una revisión de examen.

Pasaremos a explicar la sección de *Usuarios*, los elementos que pertenecen a ésta sección se muestran en la siguiente interfaz:

Sistema Integral de Evaluación de Alumnos a Distancia
Módulo de Administración

[Colegios](#) [Materias](#) [Reactivos](#) [Usuarios](#) [Cambios](#) [Menu anterior](#)

Usuario: Estatus: ACTIVO

Nombre (s):

Apellido paterno:

Apellido materno:

Correo(s):

Tipo de Usuario: ALUMNO

Contraseña:

Confirm. Contras.:

Fig. 6.20 Interfaz de la sección de usuarios

Los datos solicitados no requieren de definición, sólo se debe recordar que el usuario es el número de cuenta para el caso de un alumno, y el número de trabajador para un empleado o profesor de la institución. Tampoco olvidar que la contraseña no puede ser menor a 8 caracteres.

Como se ha mencionado, el *Tipo de Usuario* puede ser uno de tres: alumno, profesor, administrador. Cuando se asigna alguno de los últimos dos, aparecen además de los campos mostrados en la pantalla anterior, los campos y botones para poder asignar materias al usuario que se está registrando. Esto se muestra en la siguiente figura.

Nota: como se puede deducir, el proceso de asignar materias a un profesor resulta necesario para limitar al usuario en la elaboración de exámenes, pues no es aceptable que un profesor diseñara exámenes de materias que no imparte.

Sistema Integral de Evaluación de Alumnos a Distancia
Módulo de Administración

[Colegios](#) [Materias](#) [Reactivos](#) [Usuarios](#) [Cambios](#) [Menu anterior](#)

Usuario: 400091453 Estatus: ACTIVO

Nombre (s): FELIPE

Apellido paterno: AVILA

Apellido materno: VALENCIA

Correo(s): a_y_felipe@hotmail.com

Tipo de Usuario: PROFESOR

Contraseña: ●●●●●●

Confirm. Contras.: ●●●●●●

Colegio: TODOS

Materia: CALCULO

Materias Asignadas: 0

Fig. 6.21 Proceso para asignar materias a un profesor

Los campos que ahora se muestran para un usuario mayor al de tipo alumno son: *Colegio* y *Materia*, además del botón *Asignar*. El campo colegio nos permite hacer un filtro para mostrar las materias de un colegio específico en el campo *Materia*. Una vez seleccionada la materia en tal campo, la asignamos a través del botón *Asignar*. Como puede imaginar, una vez asignada una materia, dicha materia ya no estará listada en el campo *Materia* para evitar duplicar información. El resultado del proceso que acabamos de mencionar es como se muestra a continuación:

The screenshot displays the 'Módulo de Administración' interface. At the top, there are navigation tabs: Colegios, Materias, Reactivos, Usuarios, Cambios, and Menu anterior. The main area shows a form for user management. The 'Usuario' field contains '400091453' with a 'Buscar' button. The 'Estatus' is 'ACTIVO'. The user's name is 'FELIPE AVILA VALENCIA' with 'Guardar', 'Editar', and 'Nuevo' buttons. The email is 'a_v_felipe@hotmail.com'. The user type is 'PROFESOR'. The password and confirmation fields are masked with dots. The 'Colegio' is set to 'TODO' and the 'Materia' is 'CALCULO', with an 'Asignar' button. Below the form, it indicates 'Materias Asignadas: 1' and lists the assigned item: 'Clave: 1.6205', 'Materia: COMPUTACIÓN', and 'Colegio: INFORMÁTICA'. A 'Remove' button is located at the bottom.

Fig. 6.22 Resultado de asignar materias a un profesor

Se puede ver que se muestran tres datos de la materia que se ha asignado al usuario: *Clave*, *Materia* y *Colegio*, además de mostrar un número consecutivo de la materia en la lista que se mostrará en orden ascendente tomando como criterio el nombre de la materia. Además de indicarnos el número total de materias asignadas.

Para realizar el proceso inverso de asignación de materia, basta con seleccionar las materias que se desean desasociar del usuario (marcando el botón de opción múltiple que aparece del lado izquierdo a los datos de la materia en la lista) y presionar el botón *Remove*, lo que quitará de la lista de materias asignadas las que se hayan indicado.

Nota: cuando se está en modo de edición o actualización del registro, se muestran todos los datos del usuario excepto la contraseña (de hecho en modo de consulta ni siquiera aparecen los campos de *Contraseña* y *Confirmación*), esto es por razones de seguridad. Por lo que concluimos que a un usuario se le

puede actualizar su contraseña pero no se puede saber cuál es su contraseña actual.

El botón *Buscar* abre una nueva ventana en donde podemos buscar a un usuario registrado. Inicialmente, se muestran todos los usuarios de la base de datos, pero se puede hacer un filtro con los campos *Materia* y/o *Nombre* del usuario. Debe considerarse que si se especifica un valor en el campo *Materia* sólo mostrarán aquellos usuarios que tienen asignados la materia indicada. Ver la siguiente figura.



Fig. 6.23 Búsqueda de usuarios

Por terminar esta sección, mencionaremos que en la *Búsqueda de usuarios*, los botones *Nuevo* y *Cerrar*, limpian el formato en la pantalla de *Usuarios* para empezar la captura de un nuevo usuario y cierran la ventana de *Búsqueda de usuarios* respectivamente.

6.2.3 Módulo de diseño

El *Módulo de Diseño* inicia en la sección *Diseño*, cuya interfaz es la siguiente:



Fig. 6.24 Opciones en el módulo de diseño

En ésta pantalla se realiza la creación de un examen. El campo *Clave* contiene el id del examen; el id es un número único y consecutivo según se van agregando los exámenes a la base de datos, este valor es asignado por el sistema automáticamente. El campo *Materia* contiene las materias que tiene asignadas el usuario y de las cuales puede diseñar un examen. El campo colegio mostrará el colegio al que pertenece la materia seleccionada. El campo título permite capturar el nombre del examen. El campo *No. de reactivos* nos dice el número de reactivos que contiene el examen.

Cuando se selecciona una materia, se muestra un resumen de los reactivos que contiene. Observe la siguiente figura.

Sistema Integral de Evaluación de Alumnos a Distancia
Módulo de Diseño

Diseño Bitácora Menu anterior

Clave: Por asignar Guardar Buscar Diseño Aut
 Materia: COMPUTACIÓN Editar Nuevo Actualizar
 Colegio: INFORMÁTICA
 Título:

No. de reactivos: 0

Tema	Descripción	Disponibles			Total/Tema
		Básico	Intermedio	Avanzado	
1	TEMA1	1 -	3 -	2 -	6 -
1.1.	TEMA1.1	0 -	1 -	1 -	2 -
1.2.	TEMA1.2	1 -	2 -	1 -	4 -
2	TEMA2	1 -	1 -	0 -	2 -
2.1.	TEMA 2.1	1 -	1 -	0 -	2 -
3	TEMA3	1 -	3 -	3 -	7 -
3.1.	TEMA 3.1	1 -	3 -	3 -	7 -
4	TEMA4	0 -	0 -	0 -	0 -
Acumulados:		3 -	7 -	5 -	15 -

Fig. 6.25 Elementos disponibles para el diseño de examen

Como se ve en ésta pantalla, al lado derecho de cada número del resumen se encuentra un campo que servirá para indicar el número de reactivos que se desean del tema, unidad y/o temario y según el nivel: básico, intermedio y/o avanzado. Esto último aplica para el caso en que se quiere que el sistema seleccione los reactivos de forma automática. Sin embargo, para poder utilizar la selección automática, el número mínimo de reactivos a seleccionar, debe ser de al menos 5.

Nota: para poder iniciar el proceso de selección (automática o directa) de reactivos, se debe haber almacenado el examen indicando la materia y título del examen.

Una vez almacenado el examen, el botón *Diseño Aut.* se habilitará. La siguiente gráfica es un ejemplo de cómo se realiza el proceso de selección automática.

Sistema Integral de Evaluación de Alumnos a Distancia
Módulo de Diseño

Diseño Bitácora Menu anterior

Clave: 7 Guardar Buscar Diseño Aut.
 Materia: COMPUTACIÓN Editar Nuevo Actualizar
 Colegio: INFORMÁTICA
 Título: EXAMEN GUIA

No. de reactivos: 0
Disponibles

Tema	Descripción	Básico	Intermedio	Avanzado	Total/Tema
1.	TEMA1	1 -	3 -	2 - 2	6 -
1.1.	TEMA1.1	0 -	1 -	1 -	2 -
1.2.	TEMA1.2	1 -	2 - 2	1 -	4 -
2.	TEMA2	1 -	1 -	0 -	2 -
2.1.	TEMA 2.1	1 -	1 -	0 -	2 -
3.	TEMA3	1 -	3 -	3 -	7 - 2
3.1.	TEMA 3.1	1 -	3 -	3 -	7 -
4.	TEMA4	0 -	0 -	0 -	0 -
Acumulados:		3 -	7 -	5 -	15 -

Fig. 6.26 Proceso para generar un examen de forma automática

Una vez indicados los reactivos, y después de haber presionado el botón *Diseño Aut.*, el sistema selecciona los reactivos empezando siempre de adentro hacia afuera. Es decir, primero selecciona los reactivos indicados para un nivel de un tema, después, selecciona los reactivos del total de un tema, en seguida selecciona de entre los totales por nivel de una unidad, después por el total de unidad, después por el total de un nivel del temario y, finalmente, por el total del temario. Esto último es importante contemplarlo porque si, por ejemplo, considerando las condiciones de la figura anterior, se piden 2 reactivos del tema 1.2 y se indica 1 en el total del mismo tema, entonces el resultado será que se seleccionarán 2 reactivos del tema 1.2, con base en la explicación anterior.

Ahora bien, para ver los reactivos que se han seleccionado (automática o directamente) debemos hacer clic en el campo *No. de reactivos*, para la selección automática ingresada en la figura de arriba, los reactivos seleccionados fueron:

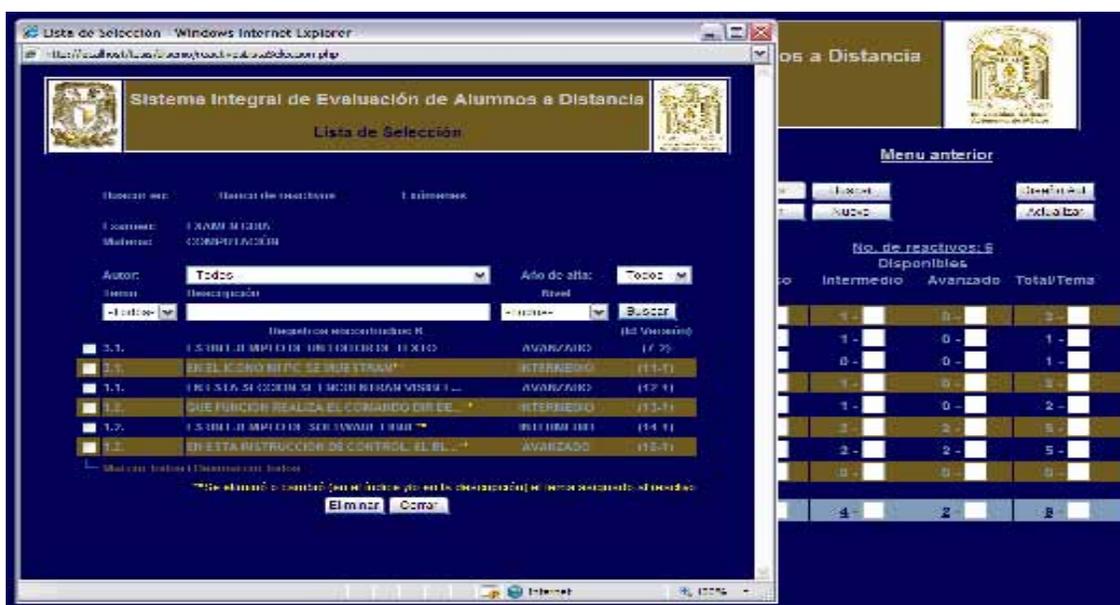


Fig. 6.27 Lista de reactivos seleccionados para el examen

Esta pantalla es muy similar a la de *Búsqueda de reactivos* explicada en la sección *Reactivos* del *Módulo de Administración*, así que sólo mencionaremos los nuevos elementos.

Nota: como se puede notar en la pantalla del fondo, *Diseño*, el resumen de reactivos ahora muestra valores diferentes en el resumen de reactivos a los mostrados antes de la selección, esto es porque los reactivos que se van agregando al examen ya no estarán disponibles para volverlos a seleccionar ni automática ni directamente.

El campo *Examen* y *Materia* nos muestran el título y la materia del examen que se está trabajando. El campo *Autor* y *Año de alta* son listas que se llenan con los diferentes autores y años de creación de los reactivos que se están mostrando, estos elementos, al igual que los campos: *Tema*, *Descripción* y *Nivel*, son criterios de búsqueda o filtro de los reactivos del examen.

A diferencia de la pantalla de *Búsqueda de reactivos*, aquí los botones que aparecen al lado izquierdo de cada reactivo, son de opción múltiple para permitir seleccionar más de uno para la operación de eliminar, es decir, para quitarlos de la lista de selección. Para ello, sólo se deben seleccionar los reactivos que se deseen remover del examen y dar clic en el botón *Eliminar*. Los campos *Marcar todos* y *Desmarcar todos* que aparecen al final de la lista de los reactivos, permiten seleccionar o quitarles la marca a todos los reactivos de la lista sin necesidad de hacerlo uno por uno.

Los campos *Banco de reactivos* y *Exámenes* nos llevan a las pantallas de búsqueda de reactivos vigentes y de exámenes anteriores de la materia a la que pertenece el examen que estamos trabajando. Veamos la interfaz de *Búsqueda en banco de reactivos*.



Fig. 6.28 Búsqueda en el banco de reactivos

Los reactivos que se muestran en esta lista son el complemento de los seleccionados para el examen, es decir, la suma de los reactivos de ésta

interfaz más los de la lista de selección, son, idealmente, el total de los reactivos de la materia a la que pertenece el examen que se está elaborando. Decimos idealmente porque puede darse el caso en que se seleccionen reactivos de un examen anterior (más adelante veremos cómo se hace ello), pero que posteriormente fueron borrados; por lo que en tal circunstancia, la suma de los reactivos de la interfaz *Búsqueda en banco de reactivos* y los de la lista de selección será mayor al total de los reactivos de la materia.

Los botones de *Agregar*, *Nuevo* y *Cerrar* agregan a la lista de selección del examen que se está trabajando los reactivos seleccionados, abren una nueva ventana para capturar un nuevo reactivo, y cierran la ventana actual respectivamente.

En este punto, veremos el detalle de un reactivo que tiene la marca para: “Se eliminó o cambió (en el índice y/o en la descripción) el tema asignado al reactivo”. Entonces veríamos algo como lo siguiente:

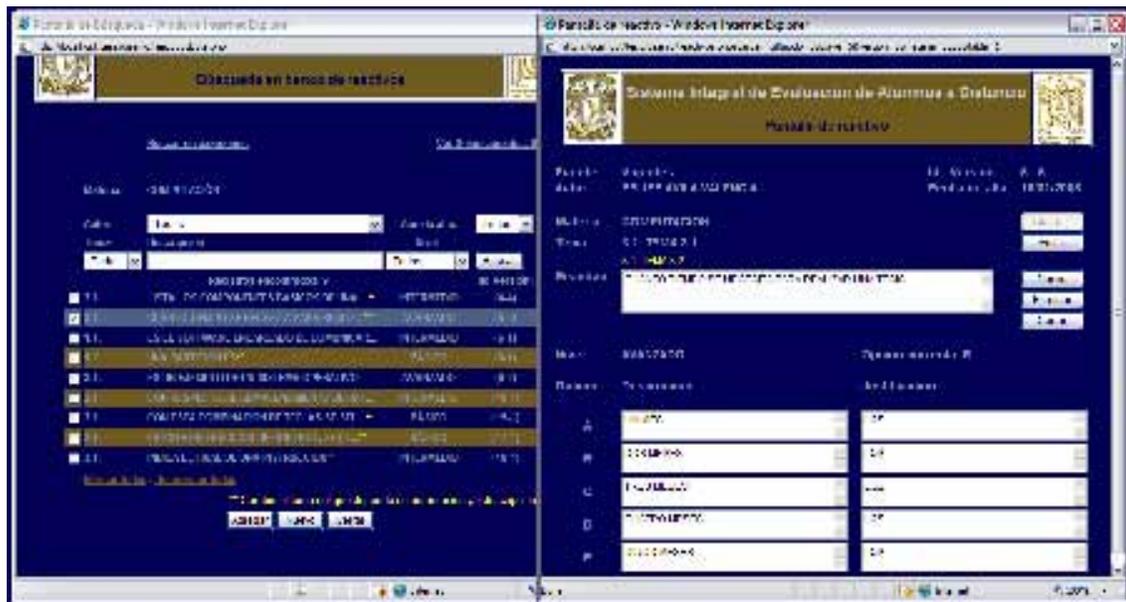


Fig. 6.29 Detalle del reactivo

Esta interfaz (*Pantalla de reactivo*) sigue teniendo las mismas funciones y opciones que la descrita en la sección de *Reactivos* del *Módulo de Administración*. El detalle de un reactivo cuyo tema del que depende ha cambiado o fue eliminado del temario (esto se cumple para ambas secciones) sólo difiere de uno que no ha cambiado su tema en la parte del campo Tema. Como se muestra en la pantalla anterior, aparece con letras amarillas el tema tal y como estaba antes de la actualización o eliminación. Así, bastará con editarlo (si se tienen los privilegios para ello, es decir, si el usuario en sesión es el autor o si es un usuario de tipo administrador, en caso contrario, el botón *Editar* y *Eliminar* estarán deshabilitados) y asignarle un tema de tal campo. Sin embargo, considere lo siguiente: a pesar de haber actualizado el tema del reactivo, el reactivo que está seleccionado para el examen sigue siendo el que tenía la versión no actualizada, por lo tanto seguirá mostrando la leyenda de que el tema del que depende ha cambiado, para cambiarlo, es necesario quitarlo de la lista y agregarlo desde la pantalla de *Búsqueda en banco de reactivos*.

Nota: cuando un usuario no tiene los privilegios necesarios para actualizar un reactivo que ha sido cambiado o eliminado el tema del que depende, más adelante se da una alternativa.

El campo *Buscar en exámenes*, que aparece en la pantalla de *Búsqueda en banco de reactivos* y en la de *Lista de selección*, nos muestra la interfaz siguiente:



Fig. 6.30 Búsqueda de reactivos en exámenes anteriores

Los campos de esta pantalla son criterios de búsqueda de reactivos en exámenes realizados por el usuario que está en sesión u otro profesor, pero en éste último caso que ya se hayan aplicado, es decir, no se tendrá acceso a exámenes de otros profesores que aunque han sido diseñados, no se han aplicado, una razón importante para que sea así es asegurar que nadie podrá ver los reactivos de un examen que no es de dominio público.

Nota: los reactivos incluidos en algún examen que no sean la versión más reciente, o que hayan sido eliminados de los reactivos vigentes, ya no podrán modificarse ni eliminarse.

En la pantalla de *Búsqueda en exámenes*, no se mostrará la lista de reactivos hasta que se indique al menos uno de los criterios de búsqueda. La siguiente imagen muestra los reactivos de los exámenes que fueron creados en el año 2008, sin olvidar que los reactivos que están en la lista de selección no estarán

disponibles para evitar volverlos a seleccionar (aunque los reactivos tengan diferentes versiones).

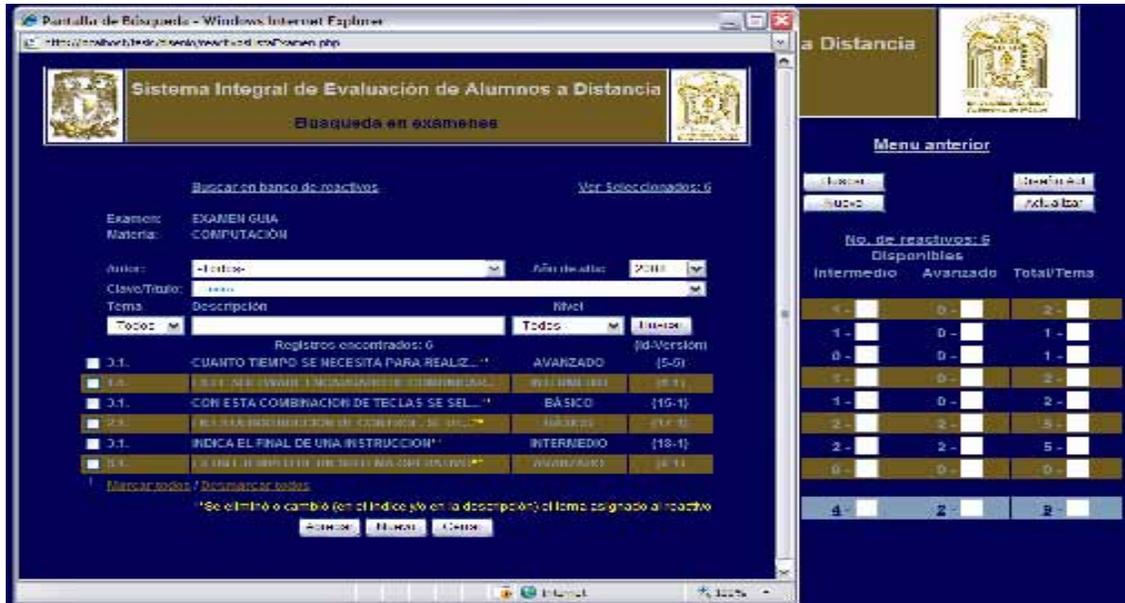


Fig. 6.31 Filtro en la búsqueda de reactivos en exámenes anteriores

Para terminar con ésta interfaz, diremos que los reactivos se agregan a nuestro examen seleccionándolos y dando clic en el botón *Agregar*. Los botones *Nuevo* y *Cerrar* abre la *Pantalla de Reactivo* para ingresar un nuevo reactivo y cierra la ventana actual respectivamente.

Finalmente, en lo que tiene que ver con la sección de diseño, diremos que el botón de *Actualizar* sirve para volver a cargar los datos de la base de datos considerando que se han modificado y/o agregado reactivos o actualizado el temario. En lo que respecta al botón *Buscar*, éste abre una ventana de búsqueda de los exámenes registrados en la base de datos y que han sido creados por el usuario en sesión, esto se cumple también para los usuarios con privilegios de administrador del sistema, para garantizar el derecho de que sólo

el autor puede modificar el examen que ha creado. A continuación se muestra esta pantalla de búsqueda.

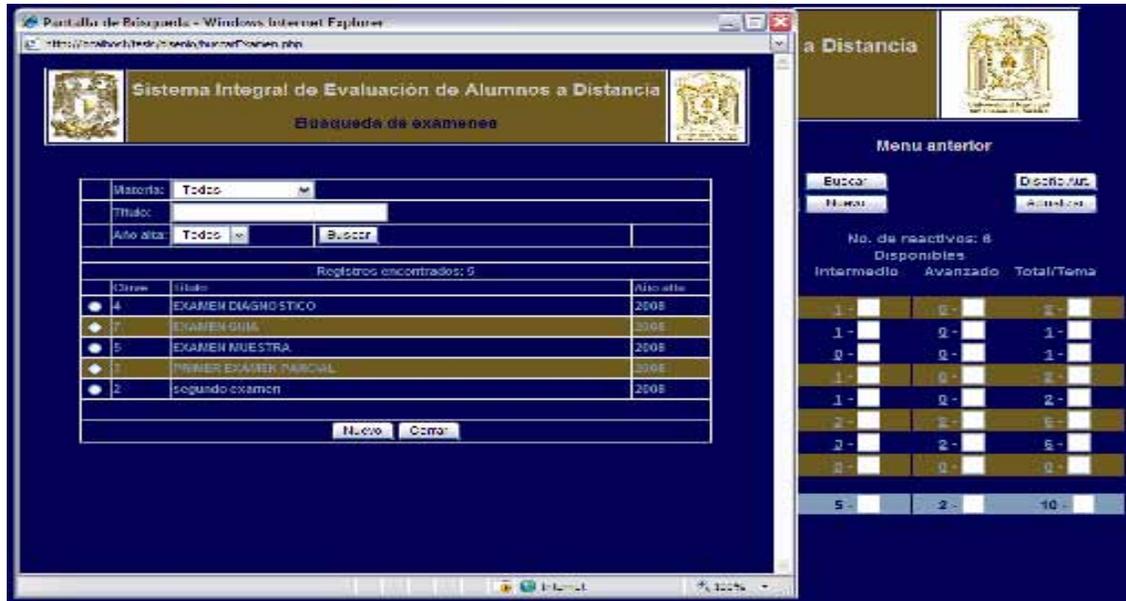


Fig. 6.32 Búsqueda de exámenes anteriores

El campo *Materia* contiene la lista de las materias que tiene asignadas el usuario en sesión. El campo *Título* y *Año de Alta* también son filtros para mostrar exámenes que cumplan con los valores proporcionados. Para cargar los datos de un examen basta con dar doble clic sobre él. Los botones *Nuevo* y *Cerrar* limpia los campos de la interfaz de diseño para iniciar un nuevo examen y cierra la ventana actual respectivamente.

Antes de iniciar la descripción de la sección *Bitácora*, presentamos la interfaz de ésta sección:



Fig. 6.33 Pantalla de la sección bitácora

El campo *Materia* lista las materias que tiene asignadas el usuario. El campo *Clave/Título* contiene los exámenes (en ese formato: clave/título) que pertenecen al usuario en sesión. El campo *Mostrar* es un filtro para mostrar las sesiones programadas del examen que se seleccione, es decir, ver las que han sido ocupadas por un usuario (un usuario presentó el examen en esa sesión), *Realizados*, o las sesiones que aún no han sido utilizadas para realizar el examen, *No realizados*.

El botón *Mostrar*, lista los tipos de sesiones indicadas del examen seleccionado. Para el desarrollo de ésta sección, se utilizará el examen creado en el desarrollo de la sección *Diseño*. Por lo tanto, no hay ninguna sesión creada para tal examen. Pero antes de crear una sesión u horario de examen, observe con detenimiento la interfaz una vez que se ha seleccionado el examen y se hace clic en el botón *Mostrar*.



Fig. 6.34 Proceso para mostrar la bitácora de un examen

Como puede apreciarse, el botón *Agregar* está deshabilitado y el campo *Validar Examen* tiene la marca para: “*No ha validado el examen*”. El sistema no permite programar un examen sino ha sido validado, al hacer clic en el campo *Validar Examen*, se abre la ventana para validar el examen. Esa ventana es la siguiente:

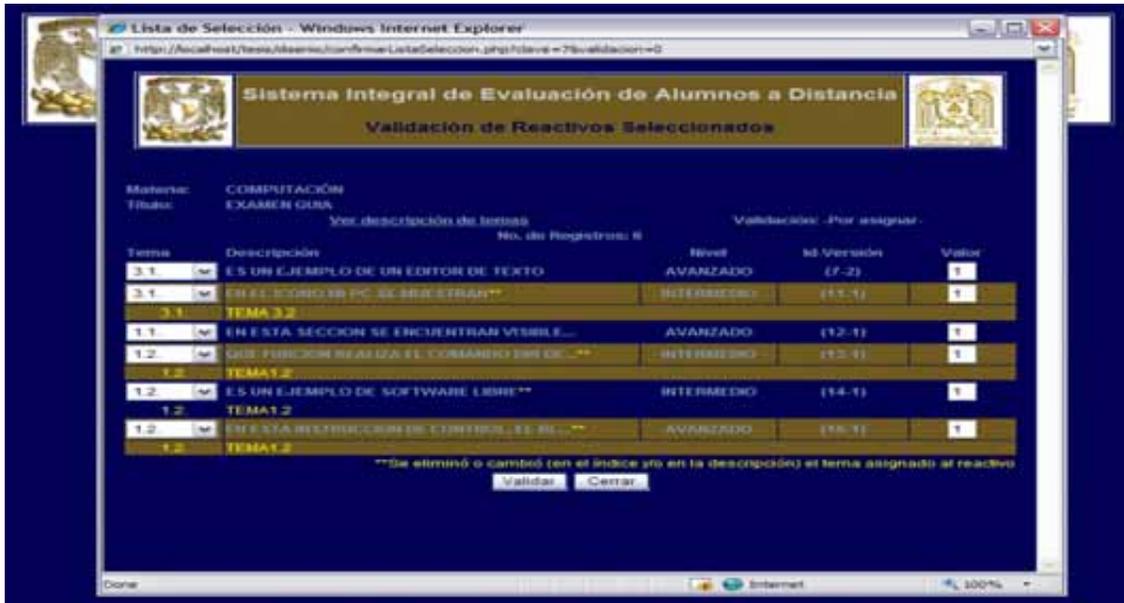


Fig. 6.35 Pantalla de validación de un examen

Hay dos razones por la que se debe validar un examen antes de poderlo programar. La primera es porque aquí es donde se asignan los puntos de cada reactivo que componen el examen. La segunda razón es porque, como se mencionó antes, si se presentó el caso en que el tema del que depende un reactivo haya sido modificado y no se tienen los privilegios para corregir tal situación, es en éste momento cuando se corrige este “problema”.

Para asignar puntos a los reactivos, que de hecho por default se les asigna valor de un punto a todos, sólo se necesita introducir el valor deseado en el campo correspondiente. Para el caso de actualizar el tema de un reactivo, basta con indicar el tema para el reactivo en el campo *Tema*.

Nota : considere que puede asignar cualquier tema a cualquier reactivo, aún si no ha cambiado el tema del que depende, esto es permitido porque en realidad no se modifica el tema del reactivo, sino sólo se hace una “máscara” para permitir mostrar en el examen en cuestión que los reactivos tienen los temas

que se decida en éste momento. Pero recuerde que los reactivos no se modifican. Otra cosa importante que se debe considerar, es que una vez que se valida un examen, no puede modificarse el diseño del mismo, es decir, quitar o agregar reactivos.

Antes de mostrar lo que se explica en la nota anterior, mencionaremos que el campo *Ver descripción de temas* abre una ventana con el temario actual de la materia, por si se desea consultar en algún momento. Observe también que el campo *Validación* tiene el valor “Por asignar”, esto es porque no se ha guardado la validación, este valor es un número consecutivo que asigna el sistema automáticamente. Esto último se hace necesario porque, una vez que un usuario realizó el examen bajo los criterios de una validación, dicha validación no podrá ser modificada y tendría que crearse una nueva validación en caso de que el temario haya sido actualizado o bien, se desee modificar el puntaje de los reactivos para poder programar una sesión del exámen. Y sólo bajo esa circunstancia se creará una nueva validación para el examen, sino es así, se podrá modificar la validación cuantas veces se desee, sin embargo, no se crea una nueva, sólo se trabaja sobre la misma validación, que se denomina *validación vigente*.

Ahora ilustremos lo explicado en la última nota. Observe la siguiente imagen que muestra la validación ya registrada en el sistema, haciendo clic en el botón *Validar*. Considere que no se puede validar un examen que tenga menos de 5 reactivos.



Fig. 6.36 Resultado de la validación de un examen

De ésta última imagen hay que resaltar los siguientes puntos. El campo validación ya tiene asignado un número que es el que se relacionará con la sesión del exámen que se programe. Los temas de los reactivos que se necesitaban actualizar se han “corregido”, pero ahora entraremos al detalle de un reactivo que en la primer imagen de la pantalla de validación estaba en tal situación. El resultado se muestra en la siguiente figura.

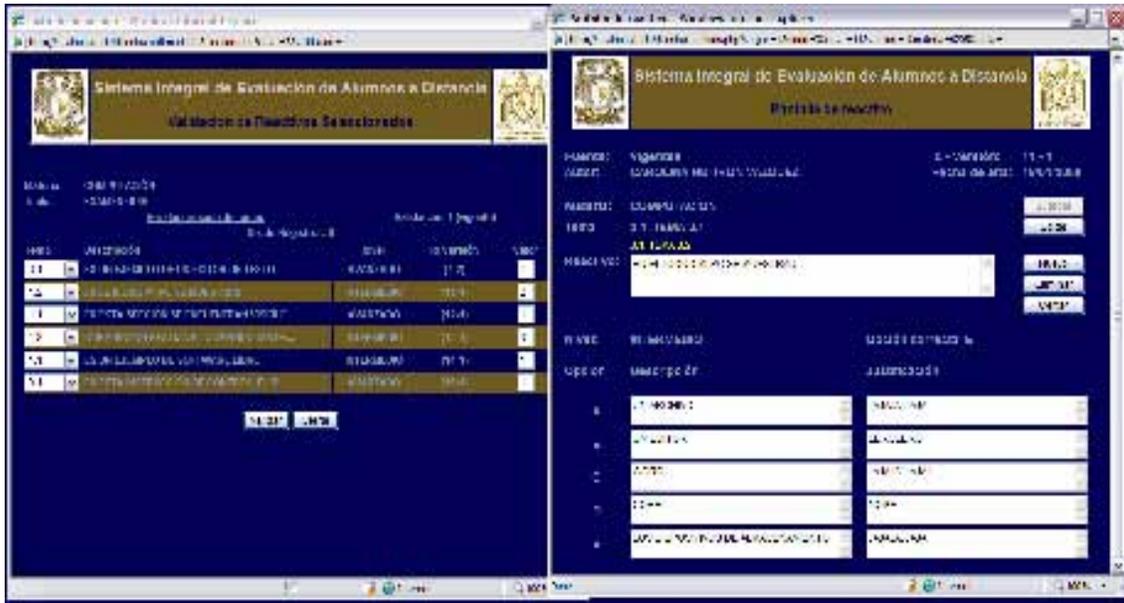


Fig. 6.37 Detalle de reactivo

De acuerdo a lo antes mencionado, el reactivo en la validación fue “asignado” al tema 1.2, sin embargo, en el detalle del reactivo, tiene asignado el tema 3.1, es decir, no fue alterado el reactivo.

Regresamos al punto de programar una sesión del examen (en la pantalla de la sección *Bitácora*); para esto, hacemos clic en el botón *Buscar* (para actualizar los datos del examen) y posteriormente en el de *Agregar*, con lo que se muestran los campos para agregar una nueva sesión. Veamos la siguiente pantalla.

Sistema Integral de Evaluación de Alumnos a Distancia
Módulo de Diseño

[Diseño](#) [Bitácora](#) [Menu anterior](#)

Materia:

Clave/Título:

Mostrar: Realizados No realizados

Validar Examen (vigente: 1)

No. Registros: 0

Id	Fecha (dd/mm/aa)	Hora (hr - min)	Duración (hr - min)	Realizaron	Validación
Por asignar	<input type="text"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	Por asignar	Por asignar

Fig. 6.38 Proceso para programar un examen

El campo *Id* es un dato importante, es un número consecutivo que asigna el sistema conforme se agregan sesiones del examen y que identifica de forma única a cada sesión. Este dato será solicitado al alumno que presentará el examen. En los campos *Fecha*, *Hora* y *Duración* se indica cuando se realizará, a qué hora y cuánto durará el examen. Los campos *Realizaron* y *Validación* son datos que el sistema proporciona, el primero es el número de alumnos que han realizado o realizaron el examen en esta sesión y, el segundo, el id o número de validación que se asignó a la sesión, esto último se hace en el momento en que el primer alumno accede al examen bajo la sesión en turno y se le asigna la validación vigente en ese momento.

Una vez mencionado lo anterior, ingresamos los datos solicitados en la pantalla de arriba y damos clic en el botón *Guardar* para registrar la sesión en la base de datos. Al guardarse, y mientras no haya ningún alumno que presentara el examen en esta sesión, estará en la lista de *No realizados* del campo *Mostrar*. Observe la siguiente imagen.



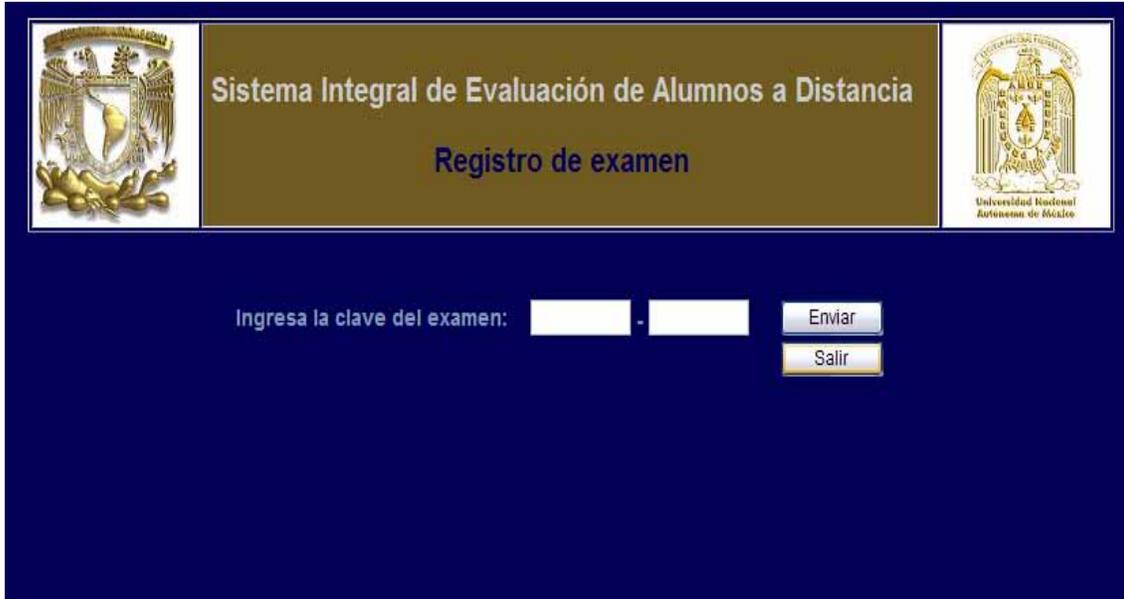
Fig. 6.39 Resultado de programar un examen

Las sesiones que no han sido utilizadas por algún alumno, pueden ser modificadas e incluso eliminadas (lo que no es posible para aquellas que pertenezcan a la lista de *Realizados*). Para modificarlas basta con actualizar los campos que se deseen y presionar el botón *Guardar*; para eliminarlas, seleccionar las sesiones deseadas y utilizar el botón *Eliminar*.

Nota: de ésta sección faltan describir los pasos para ver la lista de alumnos que han presentado el examen y la verificación del mismo por alumno. Pero esto se explicará al final del módulo de *Aplicación*.

6.2.4 Módulo de aplicación

Este módulo inicia con la interfaz *Registro de examen*, la cual se muestra a continuación.



The image shows a web interface for the 'Sistema Integral de Evaluación de Alumnos a Distancia' (SIEAD) at the 'Universidad Nacional Autónoma de México' (UNAM). The interface is titled 'Registro de examen'. It features a dark blue background with a gold header bar. The header bar contains the UNAM logo on both sides and the text 'Sistema Integral de Evaluación de Alumnos a Distancia' and 'Registro de examen' in the center. Below the header, there is a form with the label 'Ingresa la clave del examen:' followed by two input fields separated by a hyphen. To the right of the input fields are two buttons: 'Enviar' and 'Salir'.

Fig. 6.40 Pantalla para ingresar la clave del examen a realizar

La clave del examen se compone de dos datos, el id o clave del examen y el id de la sesión de tal examen a la que se desea ingresar. Estos datos son visibles todo el tiempo en la sección *Bitácora* del módulo de *Diseño*.

Nota: cuando se proporciona una clave inexistente, se ingresa una sesión que no haya iniciado aún, o bien, que haya terminado, o cuando el usuario quiere ingresar a un examen que ya fue calificado, el sistema le regresará a esta interfaz y le dará el aviso que aplique.

Cuando se ha ingresado un examen y sesión correcto y en tiempo, se nos pasa a la interfaz *Pantalla de Examen* que es en donde se contestará el examen no sin antes haber pasado por el proceso de creación del examen para el alumno, en ese proceso se ordenan de forma aleatoria los reactivos del examen y las

respuestas de cada uno para evitar que un alumno vuelva a contestar un reactivo en el mismo orden. La *Pantalla de Reactivo* se muestra en seguida.



Fig. 6.41 Interfaz de examen

Esta pantalla contiene todos los elementos para permitir al alumno contestar el examen sin necesidad de cambiar de interfaz.

Del lado izquierdo, en la parte superior, tenemos los datos generales: el nombre del profesor que diseñó el examen, el nombre del alumno (usuario en sesión), los puntos que representan la suma, 100% de los puntos totales, de los reactivos. La fecha y hora en que se ingresó, y en la que se ha programado terminará al examen, aunque para el ultimo caso, la fecha de terminación del examen por parte del alumno será real, es decir, no siempre se registrará la fecha que se programa termine el examen, cuando se al caso, la fecha y hora en la que el usuario le notifique al sistema que ha terminado de contestar el examen.

En el lado izquierdo, en la parte central, se muestra la lista con el número de reactivos del examen, así como los puntos de cada uno de ellos entre corchetes y con color amarillo. En la parte central superior, tenemos los datos de la materia, el id y título del examen, el tiempo que resta para que termine el examen (dato que se actualiza cada segundo) y, el botón *Terminar*, el cual registra al examen como terminado y nos lleva a la pantalla de calificación.

La parte central, muestra los datos del reactivo: tema al que “pertenece”, el texto del reactivo, nivel, puntos que vale (con fuente de color amarillo), y las opciones del reactivo. Además de los botones *Siguiente* y *Anterior*, que nos muestran el siguiente o anterior reactivo. Además, en la lista de los reactivos (lado izquierdo), se puede hacer clic sobre un reactivo específico para presentarlo sin necesidad de pasar uno por uno hasta llegar a él.

El funcionamiento de la pantalla es muy simple, de lado izquierdo, en la lista de reactivos, siempre estará sombreado con color café el reactivo en el que estemos ubicados. El número del reactivo, cuando no ha sido contestado es de color azul, mientras que, los reactivos que han sido contestados, estarán de color naranja. Estos elementos son de gran utilidad para ver de forma visual, el progreso en la solución del examen.

Nota: para contestar un reactivo, basta con hacer clic en el botón que aparece al lado izquierdo de la opción deseada; en ese momento, se registrará en la base de datos la respuesta. Esto es importante porque si, por alguna razón, el usuario no puede notificar que ha terminado el examen, sus respuestas ya han sido registradas. Por lo que se puede proporcionar una calificación sin problema.

Una vez que termina el examen, bien sea que el alumno notifique la terminación del mismo, o que el sistema verifique que se ha llegado a la hora de terminación, se enviará al usuario a la siguiente interfaz.

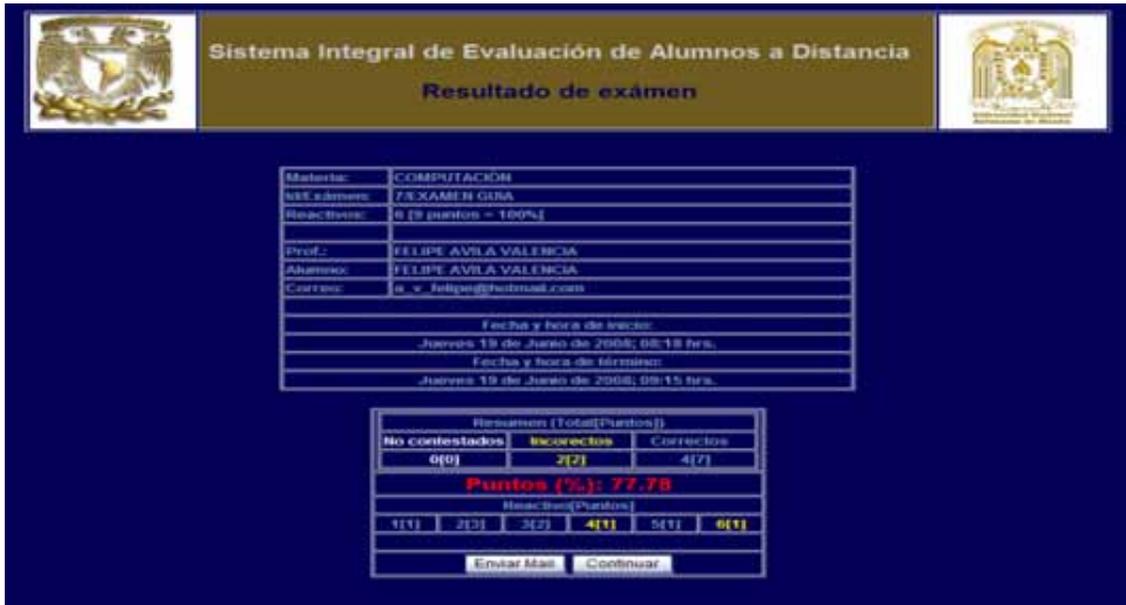


Fig. 6.42 Pantalla de resultado de examen

Esta pantalla nos muestra, además de los datos generales del examen, un resumen y detalle de la calificación del mismo. En el resumen, se tienen tres columnas, los reactivos no contestados, incorrectos y correctos de color blanco, amarillo y azul respectivamente. En esas columnas se da el total de reactivos y entre corchetes los puntos que representan. Tenemos también, el dato de *Puntos (%)*, con fuente de tamaño mayor al resto de los datos y de color rojo, que nos da el porcentaje alcanzado del 100% de los puntos. La parte de Reactivos muestra el estatus de cada reactivo utilizando los mismos colores del resumen.

El botón *Continuar* de la pantalla *Resultado de Examen*, nos lleva a la pantalla *Registro de Examen*. El botón *Enviar Mail* manda este resumen a la dirección

de correo electrónico del alumno, si no se proporcionó el botón no estará en la interfaz. Al presionar el botón para enviar el correo, se le notificará el resultado del intento al alumno. La siguiente pantalla muestra un ejemplo del correo que se envía al alumno.



Fig. 6.43 Envío de resultado de examen al correo electrónico

Nota: para poder enviar el correo, es necesario tener conexión a Internet.

Finalmente, para concluir con el desarrollo del manual de usuario, terminaremos por explicar la parte de la lista de alumnos que han presentado un examen y la revisión de examen. Para esto, tenemos que ubicarnos en el *Módulo de Diseño*, en la sección *Bitácora*. Una vez seleccionado el examen que nos interesa y mostrando las sesiones realizadas, damos clic en el número de la columna *Realizaron*. Lo que nos da la siguiente pantalla:

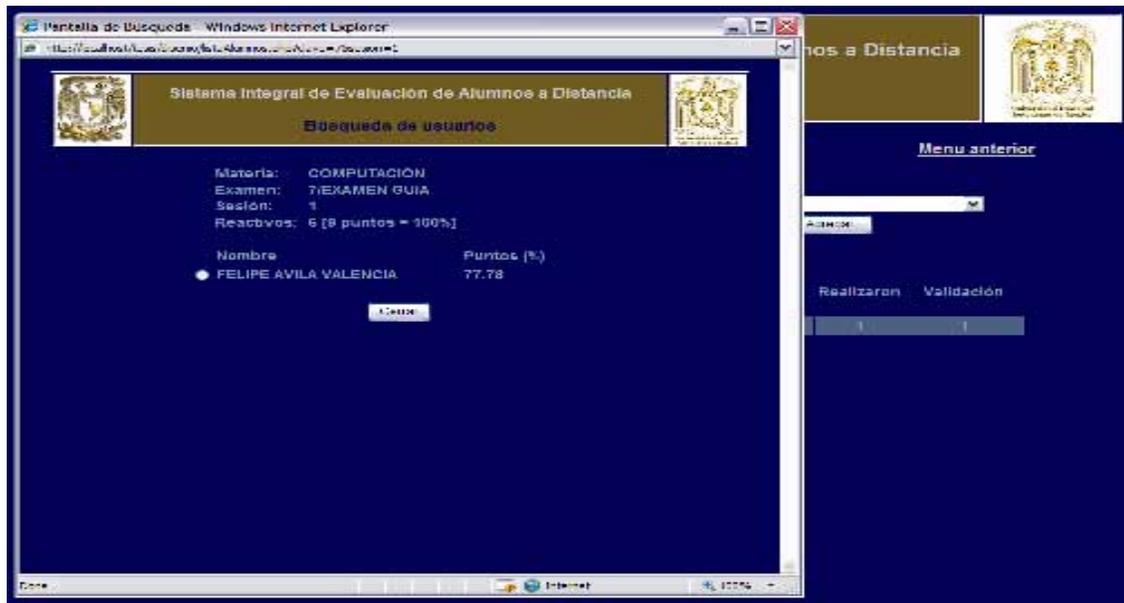


Fig. 6.44 Búsqueda de usuarios que presentaron examen

En esta pantalla se listarán todos los alumnos y el porcentaje de puntos que alcanzaron los que hayan realizado el examen en la sesión seleccionada. Para acceder a la revisión de alguno de ellos, basta con dar doble clic sobre el alumno que se desee. Básicamente verá la *Pantalla de Examen*, con la diferencia de que se mostrará los campos de justificación de las opciones del reactivo subrayada la opción elegida por el usuario y la opción que es correcta.

6.3 Pruebas del sistema

El sistema se alojó en un hosting con la intención de realizar las pruebas del sistema, el dominio que se utilizó fue: **quehayporaqui.com**; y la URL utilizada para acceder al sistema: **www://quehayporaqui.com/siead**. Nuestro asesor de tesis el, Ing. Jesús Javier Cortés Rosas reservó un laboratorio de cómputo en la División de Ciencias Básicas de la Facultad de Ingeniería el día Sábado 31 de Mayo del presente año a las 9:00 hrs.

Al laboratorio asistieron un grupo de 35 alumnos de nivel bachillerato para realizar las pruebas necesarias. Se comenzó con dos tareas concurrentes, uno de ellas fue la creación del examen que tendrían que resolver los alumnos, y con ello, todo lo que implica, registrar la materia con su temario, dar de alta los reactivos, diseñar y, finalmente, programar el examen. El examen fue formulado con 9 reactivos, y el título del mismo fue: “*Primer Examen Parcial*”. En seguida se muestran las imágenes relacionadas a los procesos anteriores.

Sistema Integral de Evaluación de Alumnos a Distancia
Módulo de Diseño

Diseño Bitácora Menu anterior

Clave:

Materia:
Colegio:
Título:

No. de reactivos: 0
Disponibles

Tema	Descripción	Básico	Intermedio	Avanzado	Total Tema
1.	CONCEPTOS BÁSICOS	0 - <input type="checkbox"/>			
1.1.	TEMA	4 - <input type="checkbox"/>	1 - <input type="checkbox"/>	3 - <input type="checkbox"/>	8 - <input type="checkbox"/>
1.2.	TEMA	0 - <input type="checkbox"/>	1 - <input type="checkbox"/>	0 - <input type="checkbox"/>	1 - <input type="checkbox"/>
1.3.	TEMA	0 - <input type="checkbox"/>			
1.4.	TEMA	0 - <input type="checkbox"/>			
1.5.	TEMA	0 - <input type="checkbox"/>			

Acumulados:

Fig. 6.45 Reactivos de la materia del examen de prueba



Fig. 6.46 Reactivos del examen de prueba

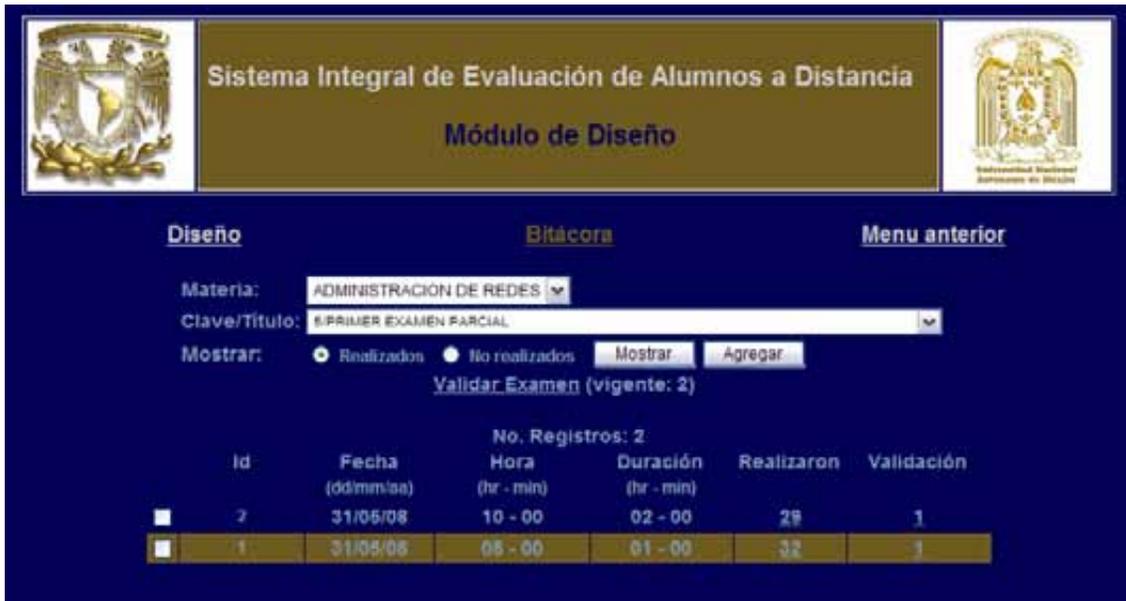


Fig. 6.47 Programación de sesiones del examen de prueba

Mientras se diseñaba el examen, se les dio indicaciones para registrarse como usuarios del sistema para que tuvieran acceso al mismo y pudieran presentar el examen que se estaba diseñando.

Después de haber sido creado el examen y de registrarse los alumnos en el sistema, se les explicó la forma de ingresar a la interfaz donde se les presentaría el examen y la forma de resolverlo. Así lo hicieron en dos ocasiones distintas, es decir, presentaron dos sesiones programadas para el examen. Esto último con la finalidad de probar la función del sistema que organiza los reactivos de forma aleatoria y sin presentar el mismo orden para las respuestas de ellos. El resultado de las dos sesiones se muestran en las siguientes imágenes.



Sistema Integral de Evaluación de Alumnos a Distancia

Búsqueda de usuarios



Materia: ADMINISTRACION DE REDES
 Examen: 5/PRIMER EXAMEN PARCIAL
 Sesión: 1
 Reactivos: 9 [9 puntos = 100%]
 No. Alumnos: 32

Usuario:	Nombre	Puntos (%)
● 040040093	ADAN MARTIN JARQUIN JARQUIN	77.78
● 050040375	ALEJANDRO ALMANZA MONROY	77.78
● 040040192	ALEJANDRO HERNANDEZ BAUTISTA	77.78
● 050040393	ANA ALEJANDRA FUENTES GUERRA	22.22
● 050040373	ANDRES ISIDRO AHEDO ROCHA	66.67
● 050040409	ARACELI AMADA JUAREZ OREA	100.00
● 050040633	CATALINA MON SERRAT VALENCIA JUAREZ	33.33
● 050040435	CECILIA RUBIO MARTINEZ	44.44
● 050040404	CHRISTIAN GONZALEZ LOPEZ	33.33
● 050040398	EDUARDO GARCIA LEON	22.22
● 050040442	EMMUEL ZAVALA DELGADO	11.11
● 400091453	FELIPE AVILA VALENCIA	33.33
● 305196969	GABRIEL ARTURO ZAMBRONI ARANDA	22.22
● 050040606	JANET YOLATL CANO SANCHEZ	22.22
● 050040439	JENNIFER EVANGELINA TOVAR CRUZ	33.33
● 050040444	JESSICA BLANCA NIETO	22.22
● 050040667	JESUS REYES RUIZ	88.89
● 050040573	JOHANNA BELEN GARCIA FLORES	11.11
● 050040380	JONATHAN ERROL CALDERON RAMOS	33.33
● 040040119	JOSE ALVAREZ AVILA	66.67
● 050040418	LILIANA NAVA GARCIA	100.00
● 050040675	MARIA GUADALUPE VAZQUEZ MORA	66.67
● 50040378	MARIA TERESA BAUTISTA CERDA	22.22
● 987654321	NANCY JANAI CHAMORRO LOZANO	22.22
● 050040714	OSCAR EDMUNDO TAPIA GRANADOS	88.89
● 050040429	PAULA FATIMA QUINTERO MARTINEZ	55.56
● 050420202	RICARDO BERMUDEZ YARZA	77.78
● 050040400	RICARDO GIJON DUQUE	22.22
● 050040552	SARA BELEN SILVESTRE ALONSO	44.44
● 050040437	SERGIO IVAN TAPIA RIVERO	22.22
● 050040419	TERESA OLGUIN CORONA	100.00
● 0400451	YESENIA ALDAMA ALARCON	77.78

Fig. 6.48 Resultado de la primera sesión del examen de prueba



Sistema Integral de Evaluación de Alumnos a Distancia

Búsqueda de usuarios



Materia: ADMINISTRACION DE REDES
 Examen: 5/PRIMER EXAMEN PARCIAL
 Sesión: 2
 Reactivos: 9 [9 puntos = 100%]
 No. Alumnos: 29

Usuario:	Nombre	Puntos (%)
040040093	ADAN MARTIN JARQUIN JARQUIN	66.67
050040375	ALEJANDRO ALMANZA MONROY	77.78
040040192	ALEJANDRO HERNANDEZ BAUTISTA	77.78
050040393	ANA ALEJANDRA FUENTES GUERRA	33.33
050040409	ARACELI AMADA JUAREZ OREA	66.67
050040633	CATALINA MONSERRAT VALENCIA JUAREZ	44.44
050040435	CECILIA RUBIO MARTINEZ	66.67
050040404	CHRISTIAN GONZALEZ LOPEZ	77.78
050040442	EMMNUEL ZAVALA DELGADO	33.33
400091453	FELIPE AVILA VALENCIA	22.22
305196969	GABRIEL ARTURO ZAMBRONI ARANDA	22.22
050040606	JANET YOLATL CANO SANCHEZ	44.44
050040439	JENNIFER EVANGELINA TOVAR CRUZ	66.67
050040444	JESSICA BLANCA NIETO	66.67
050040573	JOHANNA BELEN GARCIA FLORES	44.44
050040380	JONATHAN ERROL CALDERON RAMOS	66.67
040040119	JOSE ALVAREZ AVILA	66.67
050040418	LILIANA NAVA GARCIA	100.00
050040675	MARIA GUADALUPE VAZQUEZ MORA	100.00
50040378	MARIA TERESA BAUTISTA CERDA	44.44
987654321	NANCY JANAI CHAMORRO LOZANO	44.44
050040714	OSCAR EDMUNDO TAPIA GRANADOS	88.89
050040428	PAULA FATIMA QUINTERO MARTINEZ	66.67
050420202	RICARDO BERMUDEZ YARZA	77.78
050040400	RICARDO GIJON DUQUE	33.33
050040552	SARA BELEN SILVESTRE ALONSO	44.44
050040437	SERGIO IVAN TAPIA RIVERO	55.56
050040419	TERESA OLGUIN CORONA	100.00
0400451	YESENIA ALDAMA ALARCON	77.78

Fig. 6.49 Resultado de la segunda sesión del examen de prueba

En términos generales se cumplió con el propósito de la realización de las pruebas del sistema: ver el comportamiento de la mayoría de los procesos y funciones conforman al mismo y que permiten el diseño, programación y aplicación de exámenes. Sin embargo, se tuvieron algunos problemas por compatibilidad de versiones entre herramientas utilizadas en el desarrollo con las del servidor en el que se instaló el sistema. La mayoría se lograron corregir en ese momento, a manera de ejemplo, uno de los problemas fue el caso sensitivo en el nombre de las tablas en los scripts que ejecutaban la consulta a la base de datos; otro problema se presentó en la parte de enviar correos electrónicos, problema que no pudo ser resuelto en ese momento.

A pesar de los problemas presentados se logró verificar algo muy importante: el comportamiento y la velocidad de respuesta del sistema con el uso de múltiples usuarios. Y aunque no fue un número muy grande de usuarios concurrentes, el sistema se comportó de forma estable y no presentó problemas de lentitud. Considere para muestra de ello, que las dos sesiones se presentaron en aproximadamente 20 minutos. Esto es un buen resultado si es conocido que, entre otras cosas, las respuestas de los alumnos se registran en tiempo real, es decir, al momento de seleccionar una opción de un reactivo el sistema almacena la respuesta en la base de datos, y para ello debe ser enviada al servidor en el instante que se elige o cambia de opción. Por otro lado, el tiempo en que se resolvió el examen en ambas sesiones (sin olvidar que en la primera sesión se tomó tiempo para explicarles la forma en que debían contestar el examen, los procedimientos para ingresar y los elementos principales que se utilizaron) permite concluir que la interfaz resulta fácil de utilizar para los alumnos.

Conclusiones

Tomando en consideración el desarrollo de este documento, se puede decir que se cumplió satisfactoriamente con los requerimientos que fueron solicitados a través de la persona responsable. Entre los que se pueden destacar: un registro cabal de todas las operaciones que se ejecutan en el sistema, la confiabilidad de la información, facilidad y agilidad en el uso y comportamiento del sistema. Y aún más, la rapidez con la que, haciendo uso del sistema, se puede llevar a cabo todo el proceso de diseño, aplicación y obtención de resultados de los exámenes.

Todo lo anterior se corroboró con la realización de una serie de pruebas reales y de desarrollo que demostraron la efectividad, eficiencia y estabilidad de los procesos que el sistema realiza y que son la respuesta a las necesidades de este proyecto.

Algunos de los procesos que demandaron de una mayor inversión de tiempo en el diseño y desarrollo fueron: la generación de exámenes de forma automática que contemplara las circunstancias propias de cada alumno con la finalidad de no repetir el mismo orden de los reactivos y respuestas; otro proceso fue desarrollar un método para almacenar en tiempo real las respuestas de las preguntas planteadas.

Finalmente, otro logro que se alcanzó con la realización del sistema es la creación de un banco de datos que permitirá en un futuro no lejano, la explotación de tal información con la finalidad de verificar el proceso de evaluación y de los elementos que involucra. Este punto resulta muy trascendente si se tiene presente que se podrán tomar decisiones respaldadas en parámetros que tendrán valores confiables como resultado del análisis de la información registrada en el sistema (SIEAD).

Diccionario de datos

asignaturas

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>clave</u>	int(10)	No	0	Llave primaria que hace referencia a la clave de la asignatura, este campo es de tipo entero y almacena como máximo diez caracteres.
version	int(10)	No	0	
descripcion	varchar(100)	No		Este campo almacena el nombre de la asignatura que formara parte de la base de datos, siendo de tipo carácter con una longitud variable que puede ir desde un carácter a 255 caracteres.
colegio	int(10)	No	0	Este campo almacena el número que identifica al colegio al cual pertenece la asignatura, es de tipo entero y se define una capacidad máxima de 10 caracteres.
estatus	char(3)	No		Este campo define el estado de la asignatura si esta se encuentra activa o cancelada, es de tipo carácter y su capacidad máxima es de tres caracteres.
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en la asignatura, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dada de alta la nueva asignatura, es de tipo datetime.

cambiosasignaturas

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>clave</u>	int(10)	No	0	Llave primaria que hace referencia a la clave de la asignatura, este campo es de tipo entero y almacena como máximo diez caracteres.
<u>version</u>	int(10)	No	0	
descripcion	varchar(100)	No		Este campo almacena el nombre de la asignatura que formara parte de la base de datos, siendo de tipo carácter con una longitud variable que puede ir desde un carácter a 255 caracteres.

colegio	int(10)	No	0	Este campo almacena el número que identifica al colegio al cual pertenece la asignatura, es de tipo entero y se define una capacidad máxima de 10 caracteres.
estatus	char(3)	No		Este campo define el estado de la asignatura si esta se encuentra activa o cancelada, es de tipo carácter y su capacidad máxima es de tres caracteres.
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en la asignatura, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dada de alta la nueva asignatura, es de tipo datetime.
tipo	varchar(5)	No		Este campo almacena si los valores del cambio corresponden a una actualización o han sido eliminados, es de tipo carácter y su capacidad máxima es de tres caracteres.
usuarioCambio	varchar(20)	No		Este campo almacena los datos del usuario va a realizar el cambio de asignatura, es de tipo carácter y tiene una capacidad máxima de 20 caracteres.
fechaCambio	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual será realizado el cambio de un usuario a otra asignatura.
enviado	int(3)	No	0	Este campo se utiliza para almacenar en número de veces que ha sido enviada información por correo electrónico, es de tipo entero y su capacidad máxima es de tres dígitos.
contestado	char(2)	No		Este campo se utiliza para tener un control de los correos que hayan sido contestados, es de tipo carácter y su capacidad máxima es de dos caracteres.

cambiosColegios

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>clave</u>	int(10)	No		Llave primaria que hace referencia a la clave de colegio, este campo es de tipo entero y almacena como máximo diez caracteres.
<u>version</u>	int(10)	No	0	Llave primaria que almacena el valor numérico que indica las veces que ha sido modificado o cambiado un colegio, es de tipo entero y tiene como capacidad máxima diez dígitos.

descripcion	varchar(100)	No		Este campo almacena el nombre del colegio que forma parte de la base de datos, siendo de tipo carácter con una longitud variable que puede ir desde un carácter a 255 caracteres.
estatus	char(3)	No		Este campo define el estado del colegio si este se encuentra activo o cancelado, es de tipo carácter y su capacidad máxima es de tres caracteres.
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el colegio, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el colegio, es de tipo datetime.
tipo	varchar(5)	No		Este campo almacena si el cambio corresponde a una actualización o ha sido eliminado, es de tipo carácter y su capacidad máxima es de tres caracteres.
usuarioCambio	varchar(20)	No		Este campo almacena los datos del usuario que realizo su cambio a dicho colegio, es de tipo carácter y tiene una capacidad máxima de 20 caracteres.
fechaCambio	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual se realizo el cambio de colegio, es de tipo datetime.
enviado	int(3)	No	0	Este campo se utiliza para almacenar en número de veces que ha sido enviada información por correo electrónico, es de tipo entero y su capacidad máxima es de tres dígitos.
contestado	char(2)	No		Este campo se utiliza para tener un control de los correos que hayan sido contestados, es de tipo carácter y su capacidad máxima es de dos caracteres.

cambiosReactivos

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>id</u>	int(10)	No		Llave primaria que almacena el registro del valor que identifica al número de reactivo en particular, es de tipo entero y su capacidad máxima es de diez dígitos.
<u>version</u>	int(10)	No	0	Llave primaria que almacena el valor numérico que indica las veces que ha sido modificado o cambiado un reactivo, es de tipo entero y tiene como capacidad máxima

				diez dígitos.
idMateria	int(10)	No	0	Este campo hace referencia al número que identifica la asignatura o materia a la cual pertenece el reactivo, es de tipo entero y tiene un valor máximo de 10 dígitos.
idTema	int(10)	No	0	Este campo hace referencia al número que identifica al tema al cual pertenece el reactivo, es de tipo entero y tiene un valor máximo de 10 dígitos.
versionTema	int(10)	No	0	Este campo almacena el valor numérico que indica las veces que ha sido modificado o cambiado un tema, es de tipo entero y tiene como capacidad máxima diez dígitos.
reactivo	text	No		En este campo se almacena el texto del reactivo, es de tipo texto y su capacidad máxima es infinita.
nivel	int(3)	No	0	Este campo almacena el valor numérico que indica el grado de dificultad del reactivo, es de tipo entero y su capacidad máxima es de tres dígitos.
opcionA	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionA es de tipo texto y su longitud es infinita.
opcionB	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionB es de tipo texto y su longitud es infinita.
opcionC	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionC es de tipo texto y su longitud es infinita.
opcionD	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionD es de tipo texto y su longitud es infinita.
opcionE	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionE es de tipo texto y su longitud es infinita.
justificacionA	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionA es de tipo texto y su longitud es infinita.
justificacionB	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionB es de tipo texto y su longitud es infinita.
justificacionC	text	No		Este campo almacena el texto que justifica

				cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionC es de tipo texto y su longitud es infinita.
justificacionD	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionD es de tipo texto y su longitud es infinita.
justificacionE	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionE es de tipo texto y su longitud es infinita.
opcionCorrecta	char(1)	No		Este campo almacena la letra que identifica la opcion correcta que corresponde al reactivo, es de tipo character y su maxima longitud es de un character.
usuarioAlta	varchar(20)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el sistema, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el reactivo, es de tipo datetime.
tipo	varchar(5)	No		Este campo almacena si el cambio corresponde a una actualización o ha sido eliminado, es de tipo carácter y su capacidad es variable y va desde 0 a 255 caracteres.
usuarioCambio	varchar(20)	No		Este campo almacena los datos del usuario que realizo el cambio o modificación de un reactivo, es de tipo carácter y su longitud maxima es variable.
fechaCambio	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual se realizo el cambio del reactivo, es de tipo datetime.
enviado	int(3)	No	0	Este campo se utiliza para almacenar en número de veces que ha sido enviada información por correo electrónico, es de tipo entero y su capacidad máxima es de tres dígitos.
contestado	char(2)	No		Este campo se utiliza para tener un control de los correos que hayan sido contestados, es de tipo carácter y su capacidad máxima es de dos caracteres.

cambiosTemas

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>id</u>	int(10)	No	0	Llave primaria que almacena el registro del valor que identifica al numero del tema en particular, es de tipo entero y su capacidad máxima es de diez dígitos.
<u>version</u>	int(3)	No	0	Llave primaria que almacena el valor numérico que indica las veces que ha sido modificado o cambiado un tema, es de tipo entero y tiene como capacidad máxima tres dígitos.
claveAsignatura	int(10)	No	0	Campo que hace referencia a la clave de la asignatura, este campo es de tipo entero y almacena como máximo diez caracteres.
tema	varchar(30)	No		Este campo almacena el texto que hace referencia al tema, es de tipo carácter y su longitud es variable va desde 0 hasta 255 caracteres.
descripcion	text	No		Este campo almacena la descripción particular del tema, siendo de tipo texto con una longitud infinita de caracteres.
indice	int(3)	No	0	
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el sistema, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el tema, es de tipo datetime.
tipo	varchar(5)	No		Este campo almacena si el cambio corresponde a una actualización o ha sido eliminado, es de tipo carácter y su longitud es variable va desde 0 hasta 255 caracteres.
usuarioCambio	varchar(20)	No		Este campo almacena los datos del usuario que realizo el cambio o modificación de un tema, es de tipo carácter y su longitud es variable que va desde 0 hasta 255 caracteres.
fechaCambio	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual se realizo el cambio del tema, es de tipo datetime.
enviado	int(3)	No	0	Este campo se utiliza para almacenar en número de veces que ha sido enviada información por correo electrónico, es de tipo entero y su capacidad máxima es de tres dígitos.
contestado	char(2)	No		Este campo se utiliza para tener un control

				de los correos que hayan sido contestados, es de tipo carácter y su capacidad máxima es de dos caracteres.
--	--	--	--	--

cambiosUsuarios

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>login</u>	varchar(20)	No		Llave primaria que almacena el nombre que identifica al usuario en el sistema para poder acceder a los recursos del mismo, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
<u>version</u>	int(10)	No	0	
password	varchar(20)	No		Este campo almacena la contraseña del usuario que lo identifica para otorgarle acceso al sistema, es de tipo entero y su longitud máxima es de diez dígitos.
nombre	varchar(50)	No		En este campo se almacena el nombre propio del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
apellidoPaterno	varchar(30)	No		En este campo se almacena el apellido paterno del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
apellidoMaterno	varchar(30)	No		En este campo se almacena el apellido materno del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
correo	varchar(150)	No		Este campo almacena la dirección de correo electrónico del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres
tipoUsuario	char(3)	No		Este campo almacena el valor numérico que hace referencia al tipo de usuario del que se trata el cual puede ser usuario alumno, profesor o administrador, es de tipo carácter y su capacidad máxima es de tres dígitos.
estatus	char(3)	No		Este campo define el estado del usuario si este se encuentra activo o cancelado, es de tipo carácter y su capacidad máxima es de tres caracteres.
materias	varchar(200)	No		Este campo almacena las claves de las asignaturas a las cuales esta inscrito el usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres

usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el sistema, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el usuario, es de tipo datetime.
tipo	varchar(5)	No		Este campo almacena si el cambio corresponde a una actualización o ha sido eliminado, es de tipo carácter y su capacidad es variable y va desde 0 a 255 caracteres
usuarioCambio	varchar(20)	No		Este campo almacena los datos del usuario que realizo el cambio de un usuario en alguna asignatura, es de tipo carácter y su longitud es variable que va desde 0 hasta 255 caracteres.
fechaCambio	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual se realizaron los cambios requeridos por el usuario, es de tipo datetime.
enviado	int(3)	No	0	Este campo se utiliza para almacenar en número de veces que ha sido enviada información por correo electrónico, es de tipo entero y su capacidad máxima es de tres dígitos.
contestado	char(2)	No		Este campo se utiliza para tener un control de los correos que hayan sido contestados, es de tipo carácter y su capacidad máxima es de dos caracteres.

colegios

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>clave</u>	int(10)	No		Llave primaria que hace referencia a la clave del colegio, este campo es de tipo entero y almacena como máximo diez dígitos.
version	int(10)	No	0	Llave primaria que almacena el valor numérico que indica las veces que ha sido modificado o cambiado un colegio, es de tipo entero y tiene como capacidad máxima diez dígitos.
descripcion	varchar(100)	No		Este campo almacena el nombre del colegio que forma parte de la base de datos, siendo de tipo carácter con una longitud variable que puede ir desde un carácter a 255 caracteres.

estatus	char(3)	No		Este campo define el estado del colegio si este se encuentra activo o cancelado, es de tipo carácter y su capacidad máxima es de tres caracteres.
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el colegio, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el colegio, es de tipo datetime.

datosExamen

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>id</u>	int(10)	No		
idMateria	int(10)	No	0	Este campo hace referencia al número que identifica la asignatura o materia a la cual pertenece el examen, es de tipo entero y tiene un valor máximo de diez dígitos
titulo	text	No		Este campo describe el título del examen, es de tipo texto y su longitud es infinita.
idReactivos	text	No		Este campo almacena el registro del valor que identifica al número de reactivo en particular, es de tipo texto y su capacidad máxima es infinita.
idVersionesReactivos	text	No		Este campo almacena el valor numérico que indica las veces que ha sido modificado o cambiado un reactivo, es de tipo carácter y la longitud es variable la cual va desde 0 hasta 255 caracteres.
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario dio de alta el examen, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el examen, es de tipo datetime.

reactivos

Campo	Tipo	Valor	Valor dado	Descripción del Campo
-------	------	-------	------------	-----------------------

		nulo		
<u>id</u>	int(10)	No		Llave primaria que almacena el registro del valor que identifica al número de reactivo en particular, es de tipo entero y su capacidad máxima es de diez dígitos.
version	int(10)	No	0	Este campo almacena el valor numérico que indica las veces que ha sido modificado o cambiado un reactivo, es de tipo entero y tiene como capacidad máxima diez dígitos.
idMateria	int(10)	No	0	Este campo hace referencia al número que identifica la asignatura o materia a la cual pertenece el reactivo, es de tipo entero y tiene un valor máximo de 10 dígitos.
idTema	int(10)	No	0	Este campo hace referencia al número que identifica al tema al cual pertenece el reactivo, es de tipo entero y tiene un valor máximo de 10 dígitos.
versionTema	int(11)	No	0	Este campo almacena el valor numérico que indica las veces que ha sido modificado o cambiado un tema, es de tipo entero y tiene como capacidad máxima once dígitos.
reactivo	text	No		En este campo se almacena el texto del reactivo, es de tipo texto y su capacidad máxima es infinita.
nivel	int(3)	No	0	Este campo almacena el valor numérico que indica el grado de dificultad del reactivo, es de tipo entero y su capacidad máxima es de tres dígitos.
opcionA	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionA es de tipo texto y su longitud es infinita.
opcionB	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionB es de tipo texto y su longitud es infinita.
opcionC	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionC es de tipo texto y su longitud es infinita.
opcionD	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionD es de tipo texto y su longitud es infinita.
opcionE	text	No		Este campo almacena el texto que representa la respuesta correspondiente a la opcionE es de tipo texto y su longitud es infinita.
justificacionA	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo

				correspondiente a la opcionA es de tipo texto y su longitud es infinita.
justificacionB	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionB es de tipo texto y su longitud es infinita.
justificacionC	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionC es de tipo texto y su longitud es infinita.
justificacionD	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionD es de tipo texto y su longitud es infinita.
justificacionE	text	No		Este campo almacena el texto que justifica cada una de las opciones que pueden ser una probable respuesta para el reactivo correspondiente a la opcionE es de tipo texto y su longitud es infinita.
opcionCorrecta	char(1)	No		Este campo almacena la letra que identifica la opcion correcta que corresponde al reactivo, es de tipo carácter y su maxima longitud es de un carácter.
usuarioAlta	varchar(20)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el sistema, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No		Este campo almacena la fecha y la hora en la cual es dado de alta el reactivo, es de tipo datetime.

temas

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>id</u>	int(10)	No		Llave primaria que almacena el registro del valor que identifica al número del tema en particular, es de tipo entero y su capacidad máxima es de diez dígitos.
version	int(10)	No	0	Este campo almacena el valor numérico que indica las veces que ha sido modificado o cambiado un tema, es de tipo entero y tiene como capacidad máxima diez dígitos.
claveAsignatura	int(10)	No	0	Campo que hace referencia a la clave de la asignatura, este campo es de tipo entero y

				almacena como máximo diez caracteres.
tema	varchar(30)	No		Este campo almacena el texto que hace referencia al tema, es de tipo carácter y su longitud es variable va desde 0 hasta 255 caracteres.
descripcion	text	No		Este campo almacena la descripción particular del tema, siendo de tipo texto con una longitud infinita de caracteres.
índice	int(3)	No	0	
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el sistema, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
FechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el tema, es de tipo datetime.

usuarios

Campo	Tipo	Valor nulo	Valor dado	Descripción del Campo
<u>login</u>	varchar(20)	No		Llave primaria que almacena el nombre que identifica al usuario en el sistema para poder acceder a los recursos del mismo, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
version	int(10)	No	0	
password	varchar(20)	No		Este campo almacena la contraseña del usuario que lo identifica para otorgarle acceso al sistema, es de tipo entero y su longitud máxima es de diez dígitos.
nombre	varchar(50)	No		En este campo se almacena el nombre propio del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
apellidoPaterno	varchar(30)	No		En este campo se almacena el apellido paterno del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
apellidoMaterno	varchar(30)	No		En este campo se almacena el apellido materno del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres.
correo	varchar(150)	No		Este campo almacena la dirección de correo electrónico del usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres

tipoUsuario	char(3)	No		Este campo almacena el valor numérico que hace referencia al tipo de usuario del que se trata el cual puede ser usuario alumno, profesor o administrador, es de tipo carácter y su capacidad máxima es de tres dígitos.
estatus	char(3)	No		Este campo define el estado del usuario si este se encuentra activo o cancelado, es de tipo carácter y su capacidad máxima es de tres caracteres.
materias	varchar(200)	No		Este campo almacena las claves de las asignaturas a las cuales esta inscrito el usuario, es de tipo carácter y su longitud es variable la cual va desde 0 hasta 255 caracteres
usuarioAlta	varchar(15)	No		Este campo almacena los datos del usuario que se encuentra dado de alta en el sistema, es de tipo carácter y una de sus principales características es que su longitud es variable, tiene una capacidad máxima de 255 caracteres.
fechaAlta	datetime	No	0000-00-00 00:00:00	Este campo almacena la fecha y la hora en la cual es dado de alta el usuario, es de tipo datetime.

Índice de gráficas

Gráfica	Página
Fig. 2.1 Ciclo de vida en cascada	23
Fig. 2.2 Modelo incremental	24
Fig. 2.3 Modelo en espiral	25
Fig. 2.4 Ejemplo de un esquema cliente-servidor	28
Fig. 2.5 Ejemplo de un flujo	30
Fig. 2.6.1.a Flujo de entrada	31
Fig. 2.6.1.b Flujo de salida	31
Fig. 2.6.1.c Flujo de diálogo	31
Fig. 2.7 Representación gráfica de un almacén	32
Fig. 2.8 Representación gráfica de un terminador	33
Fig. 2.9 Funcionamiento de una aplicación PHP	42
Fig. 2.10 Esquema de un DBMS	45
Fig. 2.11 Tipos de relaciones	48
Fig. 2.12 Esquema del proceso de ejecución de una página Web con PHP	50
Fig. 4.1 Búsqueda general de claves	76
Fig. 4.2 Alta de reactivos	77
Fig. 4.3.a Modificación de reactivos	78
Fig. 4.3.b Modificación de reactivos	79
Fig. 5.1 DFD para registro de usuarios, Nivel 1	81
Fig. 5.2 DFD para validar datos, Nivel 1.2	82
Fig. 5.3 DFD para dar de alta a un usuario en el sistema, Nivel 1.3	83
Fig. 5.4 DFD para la baja de un usuario en el sistema, Nivel 1.4	84
Fig. 5.5 DFD para realizar cambios a un usuario en el sistema, Nivel 1.5	85
Fig. 5.6 Diagrama de entidad-relación	92
Fig. 5.7 Arquitectura del sistema	93
Fig. 6.1 Interfaz de EasyPHP con el menú de MySQL desplegado	96
Fig. 6.2 Interfaz de phpMyAdmin	96
Fig. 6.3 Pantalla de autenticación	98
Fig. 6.4 Mensaje para un usuario no válido	99
Fig. 6.5 Formato para registrarse como usuario del sistema	100
Fig. 6.6 Opciones disponibles para un usuario de tipo alumno	102
Fig. 6.7 Opciones disponibles para un usuario de tipo profesor	102
Fig. 6.8 Opciones disponibles para un usuario de tipo administrador	103
Fig. 6.9 Opciones del módulo de administración	104
Fig. 6.10 Modo consulta de la opción colegios	105
Fig. 6.11 Modo actualización de la opción colegios	106
Fig. 6.12 Interfaz de la opción materias	106
Fig. 6.13 Creación de un temario para una asignatura	108
Fig. 6.14 Actualización de un temario	109
Fig. 6.15 Búsqueda de asignaturas	110
Fig. 6.16 Pantalla para la sección de reactivos	111

Fig. 6.17 Resumen de reactivos de una asignatura	112
Fig. 6.18 Búsqueda de reactivos	113
Fig. 6.19 Pantalla detalle de reactivo	114
Fig. 6.20 Interfaz de la sección de usuarios	116
Fig. 6.21 Proceso para asignar materias a un profesor	117
Fig. 6.22 Resultado de asignar materias a un profesor	118
Fig. 6.23 Búsqueda de usuarios	119
Fig. 6.24 Opciones en el módulo de diseño	120
Fig. 6.25 Elementos disponibles para el diseño de examen	121
Fig. 6.26 Proceso para generar un examen de forma automática	122
Fig. 6.27 Lista de reactivos seleccionados para el examen	123
Fig. 6.28 Búsqueda en el banco de reactivos	124
Fig. 6.29 Detalle del reactivo	125
Fig. 6.30 Búsqueda de reactivos en exámenes anteriores	127
Fig. 6.31 Filtro en la búsqueda de reactivos en exámenes anteriores	128
Fig. 6.32 Búsqueda de exámenes anteriores	129
Fig. 6.33 Pantalla de la sección bitácora	130
Fig. 6.34 Proceso para mostrar la bitácora de un examen	131
Fig. 6.35 Pantalla de validación de un examen	132
Fig. 6.36 Resultado de la validación de un examen	134
Fig. 6.37 Detalle de reactivo	135
Fig. 6.38 Proceso para programar un examen	136
Fig. 6.39 Resultado de programar un examen	137
Fig. 6.40 Pantalla para ingresar la clave del examen a realizar	138
Fig. 6.41 Interfaz de examen	139
Fig. 6.42 Pantalla de resultado de examen	141
Fig. 6.43 Envío de resultado de examen al correo electrónico	142
Fig. 6.44 Búsqueda de usuarios que presentaron examen	143
Fig. 6.45 Reactivos de la materia del examen de prueba	144
Fig. 6.46 Reactivos del examen de prueba	145
Fig. 6.47 Programación de sesiones del examen de prueba	145
Fig. 6.48 Resultado de la primera sesión del examen de prueba	147
Fig. 6.49 Resultado de la segunda sesión del examen de prueba	148

Bibliografía

CENEVAL [en línea]. Disponible en: [http:// www.ceneval.edu.mx](http://www.ceneval.edu.mx) [Consulta: 24 de Noviembre de 2007]

Red Escolar SEP-ILCE [en línea]. Disponible en:
http://redescolar.ilce.edu.mx/redescolar/biblioteca/articulos/htm/fases_valor_evaluacion.htm [Consulta: 24 de Noviembre de 2007]

KOROS Ko, S. y Rossen, S. *Teaching Online. USA: Houghton Mifflin Co.* [2001, páginas 2-14]

BOOCH G. *The Future of Software Engineering.* [2000, páginas 112-123]

BOOCH G. *Análisis y Diseño Orientado a Objetos con Aplicaciones* [1996, páginas 34-51]

RUMBAUGH J. *El Lenguaje Unificado de Modelado. Manual de Referencia* [2000, páginas 45-52]

YOURDON E. *Análisis Estructurado Moderno* [1993, páginas 17-31]