



**BENEMÉRITA UNIVERSIDAD
AUTÓNOMA DE PUEBLA
FACULTAD DE CIENCIAS DE LA
COMPUTACIÓN**

**Sistema web de aplicación de pruebas objetivas para la
Licenciatura en Educación Primaria**

**Que para obtener el Título de
Licenciado en Ciencias de la Computación**

Presenta

Omar Cardoso Valle

Asesora

M.E. Ma. del Carmen Cerón Garnica

H. Puebla de Zaragoza,

CONTENIDO

INTRODUCCIÓN

CAPITULO I. MARCO TEÓRICO	1
1.1 Ingeniería de software	1
1.2 Etapas del Proceso	1
1.3 Modelos de desarrollo de software	3
1.3.1 Modelo en Cascada Clásico.....	3
1.3.2 Modelo en Espiral	5
1.3.3 Modelo de Prototipos	7
1.3.4 Desarrollo por Etapas	8
1.3.5 Desarrollo Iterativo y Creciente	8
1.4 Lenguaje Unificado de Modelado (UML)	9
1.4.1 Casos de Uso	10
1.4.2 Diagramas de Casos de Uso	11
1.4.3 Diagramas de Secuencia	11
1.4.4 Diagramas de Colaboración.....	12
1.4.5 Diagramas de Clase.....	13
1.4.6 Diseño de Componentes.....	13
1.4.7 Diagrama de Estado	13
1.4.8 Diagrama de Actividad	14
1.5 Base de Datos	14
1.5.1 Sistema de Gestión de Bases de Datos	15
1.5.2 Modelos de los Datos.....	15
1.5.2.1 Modelo Entidad – Relación	15
1.5.2.2 Modelo Relacional.....	16
1.5.3 Normalización de una Base de Datos	18
1.5.3.1 Grados de Normalización.....	18
1.6 Arquitectura Cliente – Servidor	19
1.7 Código abierto	19
1.8 Herramientas de Desarrollo	20
1.8.1 Servidor HTTP Apache	20
1.8.2 PHP	20
1.8.3 MySQL.....	21

1.8.4 Ajax.....	21
1.9 Metodología para la elaboración de reactivos	22
1.9.1 Reactivo	22
1.9.2 El reactivo de opción múltiple.....	23
CAPITULO II. ANÁLISIS DEL SISTEMA	24
2.1 Planteamiento del Problema	24
2.2 Estrategia de solución	25
2.3 Especificación de requerimientos	27
2.4 Diagramas de Casos de Uso	29
2.5 Secuencias de acciones posibles para los casos de uso del sistema	31
2.6 Especificación de Casos de Uso del sistema	32
2.7 Escenarios del sistema	40
2.8 Diagramas de Secuencia.....	44
2.8.1 Diagrama de secuencia Ingresar Docentes	44
2.8.2 Diagrama de secuencia Editar Docentes	45
2.8.3 Diagrama de secuencia Insertar Alumnos.....	45
2.8.4 Diagrama de secuencia Editar Alumnos	46
2.8.5 Diagrama de secuencia Ingresar Reactivos	46
2.8.6 Diagrama de secuencia Editar Reactivos	47
2.8.7 Diagrama de secuencia Generar Prueba	47
2.8.8 Diagrama de secuencia Asigna Prueba	48
2.8.9 Diagrama de secuencia Contestar Prueba	48
2.8.10 Diagrama de secuencia Reporte de Calificaciones	49
2.9 Diagramas de Actividades del Sistema	50
2.9.1 Diagrama de actividades Ingresar Docentes	50
2.9.2 Diagrama de actividades Modifica Docentes	50
2.9.3 Diagrama de actividades Elimina Docentes	51
2.9.4 Diagrama de actividades Ingresar alumnos	51
2.9.5 Diagrama de actividades Modifica Alumnos	52
2.9.6 Diagrama de actividades Elimina Alumnos	52
2.9.7 Diagrama de actividades Ingresar Reactivo	53
2.9.8 Diagrama de actividades Modifica Reactivo	53
2.9.9 Diagrama de actividades Elimina Reactivo	54

2.9.10 Diagrama de actividades Visualiza Docentes	54
2.9.11 Diagrama de actividades Visualiza Alumnos	55
2.9.12 Diagrama de actividades Visualiza Reactivos	55
2.9.13 Diagrama de actividades Genera Prueba	56
2.9.14 Diagrama de actividades Asigna Prueba	56
2.9.15 Diagrama de actividades Reporte de Calificaciones	57
2.9.16 Diagrama de actividades Contesta Prueba	57
CAPITULO 3. DISEÑO DEL SISTEMA	58
3.1 Arquitectura del sistema	58
3.2 Diagrama de Navegación del sistema	58
3.3 Diseño de la Base de Datos	60
3.3.1 Diseño conceptual de la Base de Datos.....	60
3.3.2 Diagrama Entidad-Relación	60
3.3.3 Diseño lógico de la Base de Datos.....	62
3.3.4 Diccionario de Datos	62
3.3.5 Diseño lógico	65
3.3.6 Normalización de la Base de Datos	79
3.3.7 Diseño físico de la Base de Datos	76
CAPITULO 4. DESARROLLO Y PRUEBAS DEL SISTEMA	79
4.1 Ambiente de desarrollo	79
4.2 Herramientas de desarrollo	79
4.3 Conexión y acceso a la base de datos del sistema	80
4.4 Requerimientos de Hardware y Software	80
4.5 Pantallas del Sistema	81
4.5.1 Pantalla de Ingreso al Sistema.....	81
4.5.2. Pantalla Menú Opciones	81
4.5.3 Pantalla Ingresar Docentes	82
4.5.4 Pantalla Ingresar Alumnos	82
4.5.5 Pantalla Docentes Registrados	83
4.5.6 Pantalla Busca Alumnos	83
4.5.7 Pantalla Acta de Nacimiento del Reactivo.....	84
4.5.8 Pantalla Busca Reactivos.....	85

4.5.9 Pantalla Edita Reactivos	85
4.5.10 Pantalla Generar Prueba.....	86
4.5.11 Pantalla Asignar Prueba.....	86
4.5.12 Pantalla Consultar Calificaciones	87
4.5.13 Pantalla Registro de Calificaciones	87
4.5.14 Pantalla Ficha de Calificación	88
4.5.15 Pantalla Ficha Gráfica resultados por grupo.....	88
4.5.16 Pantalla Impresión Resultados.....	89
4.5.17 Pantalla Prueba	89
4.6 Estadísticas de Usabilidad	90
CONCLUSIONES.....	94
PERSPECTIVAS	95
REFERENCIAS BIBLIOGRÁFICAS	96
GLOSARIO	98
ANEXO A	101

INTRODUCCIÓN

Desde hace unos años, la Secretaría de Educación Pública (SEP), aplica el Examen General de Conocimientos (EGC-LEPRI) y el examen intermedio de conocimientos (EXI-LEPRI), a los alumnos de las Escuelas Normales que cursan el sexto y cuarto semestre de la Licenciatura en Educación Primaria, Plan de estudios 1997. Evaluación que ha estado a cargo del Centro Nacional de Evaluación (CENEVAL) y que aporta información a los sustentantes, docentes y directivos de las escuelas normales y a las autoridades educativas, estatales y federales sobre el nivel de logro alcanzado por los estudiantes, con respecto a algunos conocimientos y habilidades esenciales establecidos en el plan de estudios.

Con la finalidad apoyar a que las características del egresado respondan a las exigencias del perfil de egreso de la Licenciatura en Educación Primaria y de ayudarlos a aprobar el EGC-LEPRI, EXI-LEPRI y el Examen de Oposición que presentan los egresados para el ingreso al servicio docente, se incorporó el uso de las Tecnologías de Información y Comunicaciones (TIC); creando un sistema web, que permite a la Institución Educativa, la aplicación de pruebas objetivas y la sistematización de los resultados, proporcionando a los docentes, parámetros y referencias para guiar los esfuerzos de mejoramiento educativo y la toma de decisiones de manera rápida, económica y oportuna.

El sistema permite, automatizar los procesos de, registro, reportes de resultados, la creación de un banco de reactivos, generar y aplicar de forma aleatoria pruebas objetivas para evaluar a los alumnos sobre el dominio de contenidos de una o varias asignaturas y/o de uno o varios temas.

Las pruebas se generan con preguntas seleccionadas aleatoriamente de un banco de reactivos y que constan de cuatro posibles respuestas.

Objetivo General

Elaboración de un sistema web que permita la aplicación de pruebas objetivas y la sistematización de los resultados.

Objetivos Específicos.

- Gestionar los distintos usuarios del sistema
- Gestionar preguntas del banco de reactivos
- Gestionar pruebas
- Gestionar asignación de pruebas
- Generar record de los sustentantes
- Impresión de resultados
- Mostrar de manera gráfica los resultados de un grupo o subtema

La realización de este trabajo, se basó fundamentalmente en el modelo curricular del Plan de Estudios de la Licenciatura en Educación Primaria 1997.

El sistema se desarrolló utilizando el lenguaje de programación dinámica de sitios web PHP, para el desarrollo de la bases de datos se utilizó MySQL y por último se integró HTML, AJAX y JAVASCRIPT.

El documento de tesis está organizado de la siguiente forma:

Capítulo 1. Contiene la descripción de las herramientas teóricas del desarrollo de software usadas en la aplicación.

Capítulo 2. Contiene el análisis del sistema, la especificación de requerimientos, los diagramas de casos de uso, diagrama de actividades del sistema, diagramas de secuencia y los escenarios del sistema.

Capítulo 3. Contiene la arquitectura del sistema, diagramas de navegación del sistema, así como, el diseño y modelos de la base de datos (modelo conceptual, lógico, modelo entidad-relación), la normalización y el diseño físico de la base de datos.

Capítulo 4. Contiene la descripción de las herramientas de desarrollo, las pruebas del sistema, las estadísticas de usabilidad, así como, la especificación de requerimientos de hardware y software.

Finalmente se presentan las conclusiones y perspectivas del trabajo de tesis.

CAPÍTULO 1. MARCO TEÓRICO

1.1 Ingeniería de Software

La Ingeniería del Software es una disciplina o área de la informática que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo. [1]

Comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de éste después de que se utiliza. [2]

1.2 Etapas del Proceso

La ingeniería de software requiere llevar a cabo numerosas tareas, dentro de etapas como las siguientes:

- Análisis de requerimientos
- Especificación
- Arquitectura
- Desarrollo
- Pruebas

Análisis de requerimientos: Extraer los requisitos y requerimientos de un producto de software es la primera etapa para crearlo. El resultado del análisis de requerimientos con el cliente se plasma en el documento ERS (*Especificación de Requerimientos del Sistema*). Asimismo, se define un diagrama de Entidad/Relación, en el que se plasman las principales entidades que participarán en el desarrollo del software. [3]

Especificación: La especificación de requisitos describe el comportamiento esperado en el software una vez desarrollado. Gran parte del éxito de un proyecto de software radicará en la identificación de las necesidades del negocio (definidas por la alta dirección), así como la interacción con los usuarios funcionales para la recolección, clasificación, identificación, priorización y especificación de los requisitos del software. [3]

Entre las técnicas utilizadas para la especificación de requisitos se encuentran:

- Caso de uso,
- Historias de usuario

Siendo los primeros más rigurosos y formales, los segundos más ágiles e informales. [3]

Arquitectura: La arquitectura de sistemas en general, es una actividad de planeación, ya sea a nivel de infraestructura de red y hardware, o de software. [3]

La arquitectura de software consiste en el diseño de componentes de una aplicación (entidades del negocio), generalmente utilizando patrones de arquitectura. El diseño arquitectónico debe permitir visualizar la interacción entre las entidades del negocio y además poder ser validado, por ejemplo por medio de diagramas de secuencia. Un diseño arquitectónico describe en general el cómo se construirá una aplicación de software. Para ello se documenta utilizando diagramas, por ejemplo:

- Diagramas de clases
- Diagramas de base de datos
- Diagrama de despliegue
- Diagrama de secuencia

Desarrollo: Reducir un diseño a código puede ser la parte más obvia del trabajo de ingeniería de software, pero no necesariamente es la que demanda mayor trabajo y ni la más complicada. La complejidad y la duración de esta etapa está íntimamente relacionada al o a los lenguajes de programación utilizados, así como al diseño previamente realizado. [3]

Prueba: Consiste en comprobar que el software realice correctamente las tareas indicadas en la especificación del problema. Una técnica de prueba es probar por separado cada módulo del software, y luego probarlo de forma integral, para así llegar al objetivo. [3]

Documentación: Todo lo concerniente a la documentación del propio desarrollo del software y de la gestión del proyecto, pasando por modelaciones (UML), diagramas de casos de uso, pruebas, manuales de usuario, manuales técnicos, etc; todo con el propósito de eventuales correcciones, usabilidad, mantenimiento futuro y ampliaciones al sistema. [3]

Mantenimiento: Fase dedicada a mantener y mejorar el software para corregir errores descubiertos e incorporar nuevos requisitos. Esto puede llevar más tiempo incluso que el desarrollo del software inicial. Alrededor de 2/3 del tiempo de ciclo de vida de un proyecto está dedicado a su mantenimiento. Una pequeña parte de este trabajo consiste eliminar errores (*bugs*); siendo que la mayor parte reside en extender el sistema para incorporarle nuevas funcionalidades y hacer frente a su evolución. [3]

1.3 Modelos de desarrollo de software

La ingeniería de software proporciona varios modelos o paradigmas de desarrollo en los cuales se apoya para la realización de software, de los cuales destacan los siguientes por ser los más utilizados y los más completos: [4]

- Modelo en cascada o Clásico (modelo tradicional).
- Modelo en espiral (modelo evolutivo).
- Modelo de prototipos.
- Desarrollo por etapas.
- Desarrollo iterativo y creciente o Iterativo Incremental.
- RAD (Rapid Application Development).
- Desarrollo concurrente
- Proceso Unificado
- RUP (Proceso Unificado de Rational)

A continuación se describen los primeros 5 modelos de desarrollo de software.

1.3.1 Modelo en cascada o Clásico

Es el enfoque metodológico que ordena rigurosamente las etapas del **ciclo de vida del software**, de forma tal que el inicio de cada etapa espera a la finalización de la inmediatamente anterior. [4]

Un ejemplo de una metodología de desarrollo en cascada es:

- a) Análisis de requisitos.
- b) Diseño del Sistema.
- c) Codificación.

- d) Pruebas.
- e) Mantenimiento.

De esta forma, cualquier error de diseño detectado en la etapa de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costes del desarrollo. La palabra *cascada* sugiere, mediante la metáfora de la fuerza de la gravedad, el esfuerzo necesario para introducir un cambio en las fases más avanzadas de un proyecto. [4]

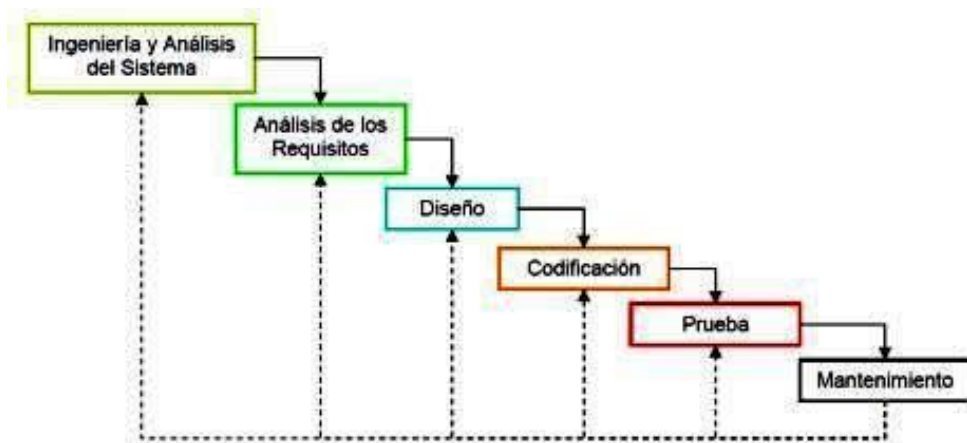


Figura 1.1 Modelo de Cascada

a) Análisis de requisitos

Se analizan las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir. De esta fase surge una memoria llamada SRD (Documento de Especificación de Requisitos), que contiene la especificación completa de lo que realizará el sistema sin entrar en detalles internos. [5]

b) Diseño del Sistema

Se descompone y organiza el sistema en elementos que se elaboran por separado, aprovechando las ventajas del desarrollo en equipo. Como resultado surge el SDD (Documento de Diseño del Software), que contiene la descripción de la estructura relacional global del sistema y la especificación de lo que realizará cada una de sus partes, así como la manera en que se combinan unas con otras. [5]

c) Diseño del Programa

Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario así como también los análisis necesarios para saber que herramientas utilizar en la etapa de Codificación. [5]

d) Codificación

Es la fase de programación o implementación propiamente dicha. Aquí se implementa el código fuente, haciendo uso de prototipos así como pruebas y ensayos para corregir errores. [5]

Dependiendo del lenguaje de programación y su versión se crean las bibliotecas y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido [5]

e) Pruebas

Los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente antes de ser puesto en explotación. [5]

f) Implantación

El software obtenido se pone en producción. Se implantan los niveles software y hardware que componen el proyecto. La implantación es la fase con más duración y con más cambios en el ciclo de elaboración de un proyecto. Es una de las fases finales del proyecto Durante la explotación del sistema, en el software tal vez sucedan cambios, bien para corregir errores o bien para introducir mejoras. Todo ello se recoge en los Documentos de Cambios. [5]

1.3.2 Modelo en Espiral.

Es un modelo de *ciclo de vida* desarrollado por Barry Boehm en 1986, utilizado generalmente en la Ingeniería de software. Las actividades de este modelo son una espiral, cada bucle es una actividad. Las actividades no están fijadas a prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior [6].

En cada vuelta o iteración hay que tener en cuenta:

- *Los Objetivos*: qué necesidad debe cubrir el producto.
- *Alternativas*: las diferentes formas de conseguir los objetivos de forma exitosa, desde diferentes puntos de vista como pueden ser:
 1. Características: experiencia del personal, requisitos a cumplir, etc.
 2. Formas de gestión del sistema.
 3. Riesgo asumido con cada alternativa.
- *Desarrollar y Verificar*: Programar y probar el software.

Para cada ciclo habrá cuatro actividades:

- Determinar Objetivos.
- Análisis del riesgo.
- Planificación.
- Desarrollar y probar.



Figura 1.2 Modelo en Espiral

Determinar o fijar objetivos

- Fijar también los productos definidos a obtener: requerimientos, especificación, manual de usuario.
- Fijar las restricciones.
- Identificación de riesgos del proyecto y estrategias alternativas para evitarlos.
- Hay una cosa que solo se hace una vez: planificación inicial.

Análisis del riesgo. Se lleva a cabo el estudio de las causas de las posibles amenazas y probables eventos no deseados y los daños y consecuencias que éstas puedan producir [6].

Planificar. Revisar todo lo hecho, evaluándolo, y con ello decidir si se continua con las fases siguientes y se planifica la próxima actividad [6].

Desarrollar, verificar y validar (probar)

- Tareas de la actividad propia y de prueba.
- Análisis de alternativas e identificación resolución de riesgos.
- Dependiendo del resultado de la evaluación de los riesgos, se elige un modelo para el desarrollo, el que puede ser cualquiera de los otros existentes, como formal, evolutivo, cascada, etc. Así si por ejemplo si los riesgos en la interfaz de usuario son dominantes, un modelo de desarrollo apropiado podría ser la construcción de prototipos evolutivos. Si lo riesgos de protección son la principal consideración, un desarrollo basado en transformaciones formales podría ser el más apropiado [6].

1.3.3 Modelo de Prototipos.

En Ingeniería de software el desarrollo con el modelo de prototipos o modelo de desarrollo evolutivo, se inicia con la definición de los objetivos globales para el software, luego se identifican los requisitos conocidos y las áreas del esquema en donde es necesaria más definición. Se utiliza para dar al usuario una vista preliminar de parte del software. Este modelo es básicamente prueba y error ya que si al usuario no le gusta una parte del prototipo significa que la prueba fallo por lo cual se debe corregir el error que se tenga hasta que el usuario quede satisfecho [7].



Figura 1.3 Modelo de Prototipos

El diseño rápido se centra en una representación de aquellos aspectos del software que serán visibles para el cliente o el usuario final (por ejemplo, la configuración de la interfaz con el usuario y el formato de los despliegues de salida). El diseño rápido conduce a la construcción de un prototipo, el cual es evaluado por el cliente o el usuario para una retroalimentación; gracias a ésta se refinan los requisitos del software que se desarrollará.

La iteración ocurre cuando el prototipo se tiene que ajustar para satisfacer las necesidades del cliente.

Esto permite que el desarrollador entienda mejor lo que se realizará y el cliente vea resultados a corto plazo.

1.3.4 Desarrollo por Etapas.

El modelo de desarrollo de software por etapas es similar al Modelo de prototipos ya que se muestra al cliente el software en diferentes estados sucesivos de desarrollo, se diferencia en que las especificaciones no son conocidas en detalle al inicio del proyecto y por tanto se van desarrollando simultáneamente con las diferentes versiones del código [8].

Se distinguen las siguientes fases:

- Especificación conceptual
- Análisis de requerimientos
- Diseño inicial
- Diseño detallado, codificación, depuración y liberación

Estas diferentes fases se van repitiendo en cada etapa del diseño.

1.3.5 Desarrollo Iterativo y Creciente.

Es un proceso de desarrollo de software, creado en respuesta a las debilidades del modelo tradicional de cascada. Para apoyar el desarrollo de proyectos por medio de este modelo se han desarrollado entornos de trabajo, de los cuales los dos más famosos son el Rational Unified Process (RUP) y el Dynamic Systems Development Method (DSDM). El desarrollo incremental e iterativo es también una parte esencial de un tipo de programación conocido

como programación extrema y los demás entornos de trabajo de desarrollo rápido de software [9].

La idea principal detrás del mejoramiento iterativo es desarrollar un sistema de programas de manera incremental, permitiéndole al desarrollador sacar ventaja de lo que se ha aprendido a lo largo del desarrollo anterior, incrementando, versiones entregables del sistema. El aprendizaje viene de dos vertientes: el desarrollo del sistema y su uso.

1.4 Lenguaje Unificado de Modelado (UML)

Lenguaje de modelado de sistemas de software que se ha convertido en estándar en el mundo del desarrollo de sistemas. Es el resultado del trabajo hecho por Grady Booch, James Rumbaugh e Ivar Jacobson y está respaldado por el OMG (Object Management Group). Está constituido por un conjunto de diagramas, y proporciona un estándar que permite al analista de sistemas generar un anteproyecto de varias facetas que sean comprensibles para los clientes, desarrolladores y todos aquellos que están involucrados en el proceso de desarrollo. [10]

Se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. Ofrece nueve diagramas en los cuales modelar sistemas.

- Diagramas de Casos de Uso para modelar los procesos.
- Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
- Diagramas de Colaboración para modelar interacciones entre objetos.
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- Diagramas de Componentes para modelar componentes.
- Diagramas de Implementación para modelar la distribución del sistema.

La finalidad de los diagramas, es presentar diversas perspectivas de un sistema, a los cuales se les conoce como modelo. Es importante destacar que un modelo UML describe lo que hará un sistema, pero no dice cómo implementar dicho sistema.

1.4.1 Casos de Uso

Es una estructura para describir la forma en que un sistema lucirá para los usuarios potenciales. Es una colección de escenarios iniciados por una entidad llamada actor (una persona, un componente de hardware, un lapso u otro sistema). Un caso de uso debería dar por resultado algo de valor ya sea para el actor que lo inició o para otro. Es un escenario que describe cómo el software va a ser usado en una determinada situación.

Los Casos de Uso no son parte del diseño (el cómo), sino parte del análisis (el qué). De forma que al ser parte del análisis nos ayudan a describir qué es lo que es sistema debe hacer. Los Casos de Uso describen un uso del sistema y cómo este interactúa con el usuario. Son generalmente el punto de partida del análisis orientado a objetos con UML. [11]

Para crear un caso de uso, el analista debe primero identificar los diferentes tipos de personas (o dispositivos) que utiliza el sistema o producto. Estos *actores* actualmente representan papeles que la gente (o dispositivos) juegan como impulsores del sistema. Definido más formalmente, un actor es algo que comunica con el sistema o producto y que es externo al sistema en sí mismo.

El objetivo es construir un Diagrama de Caso de Uso para cada tipo de escenario diferente en el sistema. Cada escenario muestra una secuencia diferente de interacciones entre actores y el sistema, sin condiciones 'or'.

Jacobson [JAC93] sugiere un número de preguntas que deberán responderse por el caso de uso:

- ¿Cuáles son las principales tareas o funciones que serán realizadas por el actor?
- ¿Cuál es el sistema de información que el actor adquiere, produce o cambia?
- ¿Qué actor informará al sistema de los cambios en el entorno externo?
- ¿Qué información necesita el actor sobre el sistema?

1.4.2 Diagramas de casos de uso

En estos se explica gráficamente un conjunto de casos de uso de un sistema, los actores y la relación entre éstos. Tiene por objeto ofrecer una clase de diagrama contextual que nos permite conocer rápidamente los actores externos de un sistema y las formas básicas en que lo utilizan. [10]

Imagínese al caso de uso como una colección de situaciones respecto al uso de un sistema. Cada escenario describe una secuencia de eventos. Cada secuencia se inicia por una persona, otro sistema, una parte del hardware o por el paso del tiempo. El resultado de la secuencia debe ser utilizable por el actor que la inició, o por otro actor.

Es importante indicar que un actor y un usuario no son la misma cosa. Un usuario normal puede jugar un número de papeles diferentes cuando utiliza un sistema, por lo tanto un actor representa una clase de entidades externas (a veces, pero no siempre personas) que lleva a cabo un papel. Como ejemplo, considerar un operador de una máquina (un usuario) que interactúa con el ordenador central para un elemento de fabricación que contiene un número de robots y máquinas bajo control numérico.

1.4.3 Diagramas de secuencia

Es uno de los diagramas más efectivos para modelar interacción entre objetos en un sistema. Un diagrama de secuencia se modela para cada uso. Mientras que el diagrama de caso de uso permite el modelado de una vista 'business' del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes pasados entre los objetos.

Muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como vectores horizontales. Los mensajes se dibujan cronológicamente desde la parte superior del diagrama a la parte inferior; la distribución horizontal de los objetos es arbitraria. [12]

Un diagrama de secuencia muestra la forma en que los objetos se comunican entre sí al transcurrir el tiempo. Consta de objetos que se representan del modo usual: rectángulos

con nombre (subrayado), mensajes representados por líneas continuas con una punta de flecha y el tiempo representado como progresión vertical.

Objetos: Se colocan cerca de la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La extensión que está debajo (y en forma descendente) de cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Junto a la línea de vida de un objeto se encuentra un pequeño rectángulo conocido como *activación*, el cual representa la ejecución de una operación que realiza el objeto. La longitud del rectángulo se interpreta como la duración de la activación.

Mensaje: Un mensaje que va de un objeto a otro pasa de la línea de vida de un objeto a la de otro. Un objeto puede enviarse un mensaje a sí mismo. Puede ser simple, sincrónico, o asincrónico.

Un mensaje simple es la transferencia del control de un objeto a otro. Si un objeto envía un mensaje sincrónico, esperará la respuesta a tal mensaje antes de continuar con su trabajo. Si un objeto envía un mensaje asincrónico, no esperará una respuesta antes de continuar. En el diagrama de secuencia, los símbolos del mensaje varían, por ejemplo, la punta de la flecha de un mensaje simple está formada por dos líneas, la punta de la flecha de un mensaje sincrónico está rellena y la de un asincrónico tiene una sola línea, como se aprecia en la figura.

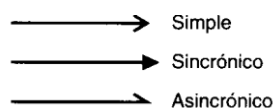


Figura 1.4 Ejemplos de Mensaje

1.4.4 Diagrama de colaboración

El diagrama de colaboración presenta una alternativa al diagrama de secuencia para modelar iteraciones entre objetos en el sistema. Mientras que el diagrama de secuencias se centra en la secuencia cronológica del escenario que estamos modelando, el diagrama de colaboración se centra en estudiar todos los efectos de un objeto dado durante un escenario. Los objetos se conectan por medio de enlaces, cada enlace representa una instancia de una asociación entre las clases implicadas. El enlace muestra los mensajes enviados entre los

objetos, el tipo de mensaje (sincrónico, asincrónico, simple, blanking, y 'time-out'), y la visibilidad de un objeto con respecto a los otros. [12]

1.4.5 Diagrama de clase

El Diagrama de Clase es el diagrama principal de diseño y análisis para un sistema. En él, la estructura de clases del sistema se especifica, con relaciones entre clases y estructuras de herencia. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal. Durante el diseño, se usa el mismo diagrama, y se modifica para satisfacer los detalles de las implementaciones. [12]

El diagrama de clases se desarrolla a través de información obtenida en los Casos de Uso, Diagramas de Secuencia y Diagramas de Colaboración. Los objetos encontrados durante el análisis son modelados en términos de la clase a la que instancian, y las interacciones entre objetos son referenciados a relaciones entre las clases instanciadas. Durante el diseño, el diagrama de clase se elabora para tener en cuenta los detalles concretos de la implementación del sistema.

1.4.6 Diseño de Componentes

Un componente es un grupo de objetos o componentes más pequeños que interactúan entre ellos y se combinan para dar un servicio. Un componente es similar a una caja negra, en la cual los servicios del componente se especifican por su interface o interfaces, sin ofrecer conocimiento del diseño e implementación internas del componente. El desarrollo basado en componentes es el proceso de ensamblar la combinación correcta de componentes en la configuración correcta para llevar a cabo la funcionalidad deseada para un sistema. Los componentes se representan en el diagrama de clases de UML especificando la interfaz de una clase o paquete. [12]

1.4.7 Diagrama de estado

Mientras los diagramas de interacción y colaboración modelan secuencias dinámicas de acción entre grupos de objetos en un sistema, el diagrama de estado se usa para modelar el comportamiento dinámico de un objeto en particular, o de una clase de objetos.

Un diagrama de estado se modela para todas las clases que se consideran con un comportamiento dinámico. En él, se modela la secuencia de estado que un objeto de la clase atraviesa durante su vida en respuesta a los estímulos recibidos, junto con sus propias respuestas y acciones. [12]

Los estados representan las condiciones de objetos en ciertos puntos en el tiempo. Los eventos representan incidentes que hacen que los objetos pasen de un estado a otro. Las líneas de transición describen el movimiento desde un estado hasta otro. Cada línea de transición se nombre con el evento que causa esta transición. Las acciones ocurren cuando un objeto llega a un estado.

1.4.8 Diagramas de Actividad

El Diagrama de Actividad es un diagrama de flujo del proceso multi-propósito que se usa para modelar el comportamiento del sistema. Los diagramas de actividad se pueden usar para modelar un Caso de Uso, o una clase, o un método complicado.

Un diagrama de actividad es parecido a un diagrama de flujo; la diferencia clave es que los diagramas de actividad pueden mostrar procesado paralelo (parallel processing). Esto es importante cuando se usan diagramas de actividad para modelar procesos 'business' algunos de los cuales pueden actuar en paralelo, y para modelar varios hilos en los programas concurrentes. [12]

1.5 Base de Datos

Las bases de datos forman una parte esencial de casi todas las empresas actuales. A lo largo de las últimas cuatro décadas del siglo veinte, el uso de las bases de datos creció en todas las empresas. En los primeros días, muy pocas personas interactuaron directamente con los sistemas de bases de datos, aunque sin darse cuenta interactuaron con bases de datos indirectamente (con los informes impresos como extractos de tarjetas de crédito, o mediante agentes como cajeros de bancos y agentes de reserva de líneas aéreas). Después vinieron los cajeros automáticos y permitieron a los usuarios interactuar con las bases de datos. Las interfaces telefónicas con los computadores (sistemas de respuesta vocal interactiva) también permitieron a los usuarios manejar directamente las bases de datos. [13]

1.5.1 Sistema de Gestión de Bases de Datos

Un sistema gestor de bases de datos (SGBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada **base de datos**, contiene información relevante para una empresa. El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto *práctica* como *eficiente*.

La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información. Además, los sistemas de bases de datos deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o los intentos de acceso sin autorización. Si los datos van a ser compartidos entre diversos usuarios, el sistema debe evitar posibles resultados anómalos. [13]

1.5.2 Modelos de los Datos

Una colección de herramientas conceptuales para describir los datos, las relaciones, la semántica y las restricciones de consistencia. Estos se clasifican en tres grupos diferentes: modelos lógicos basados en objetos, modelos lógicos basados en registros y modelos físicos [13].

Para ilustrar el concepto describiremos dos modelos de datos: el modelo entidad relación y el modelo relacional.

1.5.2.1 Modelo entidad-relación

Está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados *entidades*, y de *relaciones* entre estos objetos. Una entidad es una <<cosa>> u <<objeto>> en el mundo real que es distinguible de otros objetos. [13]

Las entidades se describen en una base de datos mediante un conjunto de **atributos**.

Una relación es una asociación entre varias entidades. El conjunto de todas las entidades del mismo tipo, y el conjunto de todas las relaciones del mismo tipo, se denominan respectivamente conjunto de entidades y conjunto de relaciones [Abraham Silberschatz].

La estructura lógica general de una base de datos se puede expresar gráficamente mediante un *diagrama E - R*, que consta de los siguientes componentes:

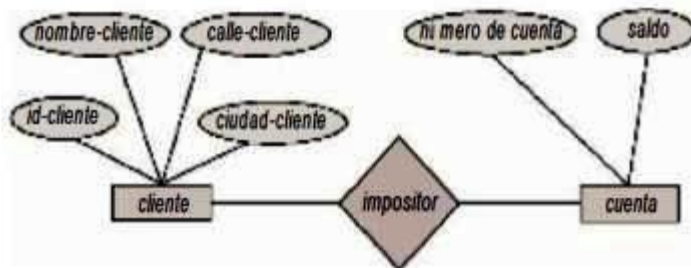
Rectángulos: Representan conjuntos de entidades.

Elipses: Representan atributos.

Rombos: Representan relaciones entre conjuntos de entidades.

Líneas: Unen los atributos con los conjuntos de entidades y los conjuntos de entidades con las relaciones.

Cada componente se etiqueta con la entidad o relación que representa.



1.5 Ejemplo de diagrama E- R

1.5.2.2 Modelo relacional

Se ha establecido actualmente como el principal modelo de datos para las aplicaciones de procesamiento de datos debido a su simplicidad, que facilita el trabajo del programador en comparación con otros modelos anteriores como el de red y el jerárquico. [Abraham Silberschatz]

En el modelo relacional se utiliza un grupo de tablas para representar los datos y las relaciones entre ellos. Cada tabla está compuesta por varias columnas, y cada columna tiene un nombre único. En la figura 1.6 se presenta un ejemplo de base de datos relacional consistente en tres tablas.

<i>id-cliente</i>	<i>nombre-cliente</i>	<i>calle-cliente</i>	<i>ciudad-cliente</i>
19.283.746	González	Arenal	La Granja
01.928.374	Gómez	Carretas	Cerceda
67.789.901	López	Mayor	Peguerinos
18.273.609	Abril	Preciados	Valsain
32.112.312	Santos	Mayor	Peguerinos
33.666.999	Rupérez	Ramblas	León
01.928.374	Gómez	Carretas	Cerceda

(a) La tabla *cliente*

<i>nº mero-cuenta</i>	<i>saldo</i>
C-101	500
C-215	700
C-102	400
C-305	350
C-201	900
C-217	750
C-222	700

(b) La tabla *cuenta*

<i>id-cliente</i>	<i>nº mero-cuenta</i>
19.283.746	C-101
19.283.746	C-201
01.928.374	C-215
67.789.901	C-102
18.273.609	C-305
32.112.312	C-217
33.666.999	C-222
01.928.374	C-201

(b) La tabla *impositor*

FIGURA 1.6. Ejemplo de base de datos relacional.

Este es el modelo de datos más ampliamente usado, y una amplia mayoría de sistemas de bases de datos actuales se basan en el modelo relacional.

El modelo relacional se encuentra a un nivel de abstracción inferior al modelo E-R. Los diseños de bases de datos a menudo se realizan en el modelo E-R, y después se traducen al modelo relacional. [13]

Llaves Primarias y Foráneas.

Este tipo de llaves determinan la forma en cómo se relacionan entre si las distintas tablas que forman el modelo entidad relación.

Llave primaria. Es un atributo o conjunto de atributos que permiten identificar de manera única cada fila en una relación.

Llave foránea. Es un atributo o conjunto de atributos en una relación que hace referencia a la llave primaria de otra relación para poder establecer una relación [6].

1.5.3 Normalización de una Base de Datos

La normalización es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más pequeñas, que además de ser más simples y más estables, son más fáciles de mantener. También se puede entender la normalización como una serie de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar un esquema que minimice los problemas de lógica. Cada regla está basada en la que le antecede. La normalización se adoptó porque el viejo estilo de poner todos los datos en un solo lugar, como un archivo o una tabla de la base de datos, era ineficiente y conducía a errores de lógica cuando se trataban de manipular los datos.[15]

Una ventaja de la normalización de base de datos, es el consumo de espacio. Una base de datos normalizada ocupa menos espacio en disco que una no normalizada. Hay menos repetición de datos, lo que tiene como consecuencia un mucho menor uso de espacio en disco. [15]

1.5.3.1 Grados de normalización

Existen básicamente tres niveles de normalización: Primera Forma Normal (1NF), Segunda Forma Normal (2NF) y Tercera Forma Normal (3NF). Cada una de estas formas tiene sus propias reglas.

Primera Forma Normal: La regla de la Primera Forma Normal establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.

Poner la base de datos en la Primera Forma Normal resuelve el problema de los encabezados de columna múltiples. Muy a menudo, los diseñadores de bases de datos inexpertos harán algo similar a la tabla no normalizada. Una y otra vez, crearán columnas que representen los mismos datos. La normalización ayuda a clarificar la base de datos y a organizarla en partes más pequeñas y más fáciles de entender. En lugar de tener que entender una tabla gigantesca y monolítica que tiene muchos diferentes aspectos, sólo tenemos que entender los objetos pequeños y más tangibles, así como las relaciones que guardan con otros objetos también pequeños.

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

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

Tercera Forma Normal: Una tabla está normalizada en esta forma si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Comentamos anteriormente que una dependencia transitiva es aquella en la cual existen columnas que no son llave que dependen de otras columnas que tampoco son llave.

Cuando las tablas están en la Tercera Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no debe haber datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir. [15]

1.6 Arquitectura Cliente Servidor

- Un sistema de base de datos puede ser visto como un sistema que tiene una estructura muy sencilla de dos partes, las cuales consisten en un servidor y un conjunto de clientes. [16]
- El servidor es precisamente el propio DBMS. Soporta todas las funciones básicas del DBMS (definición de datos, manipulación de datos, seguridad e integridad de los datos, etc.).
- Los clientes son las diversas aplicaciones que se ejecutan sobre el DBMS, tanto aplicaciones escritas por el usuario como aplicaciones integradas [16]

1.7 Código abierto

Es el término con el que se conoce al software distribuido y desarrollado libremente.

1.8 Herramientas de Desarrollo

Las herramientas de desarrollo son aquellos programas o aplicaciones que tengan cierta importancia en el desarrollo de un programa (programación). Pueden ser de importancia vital (como un ensamblador, un compilador o un editor) o de importancia secundaria, como una IDE (Integrated Development Environment - Entorno de Desarrollo Integrado). [25]

1.8.1 Servidor HTTP Apache

Es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web. [17]

Es el componente de servidor web en la popular plataforma de aplicaciones LAMP, junto a MySQL y los lenguajes de programación PHP/Perl/Python (y ahora también Ruby). Es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. Un ejemplo es al momento de compartir archivos desde una computadora personal hacia Internet. Un usuario que tiene Apache instalado en su escritorio puede colocar arbitrariamente archivos en la raíz de documentos de Apache, desde donde pueden ser compartidos. Los programadores de aplicaciones web a veces utilizan una versión local de Apache con el fin de previsualizar y probar código mientras éste es desarrollado.[17]

1.8.2 PHP

PHP es un lenguaje de programación usado normalmente para la creación de contenido para sitios web con los cuales se puede programar las páginas html y los códigos de fuente. PHP es un acrónimo recursivo que significa "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, Personal Home Page Tools), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web.

PHP es un lenguaje interpretado basado principalmente en C, C++ y Java, con los que comparte prácticamente toda su sintaxis y semántica, y aporta también algunas características de lenguajes interpretados como Perl y Bash. [18]



Figura 1.7 Ejemplos de Mensaje

Su interpretación y ejecución se da en el servidor web, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente.

1.8.3 MySQL

El gestor de bases de datos MySQL se basa en el sistema de bases de datos relacionales, conocido por las siglas RDBMS (Relational Database Management System). Para poder establecer su gestión, el servidor MySQL utiliza un lenguaje propio interno basado en estructuras de consultas. Este lenguaje es conocido con el nombre de SQL (Structured Query Language) o lenguaje de consultas estructuradas. El estándar SQL adopta el modelo ANSI, basado en una normalización norteamericana. [20]

1.8.4 AJAX

Acronónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano.

De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado (scripting language) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante *XMLHttpRequest*, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML. Es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM). [21]

1.9 Metodología para la elaboración de reactivos

Elaborar reactivos para un examen estandarizado requiere de una metodología precisa que nos permita obtener reactivos estadísticamente confiables, capaces de discriminar entre los alumnos de alto y bajo rendimiento. De esta forma, nos darán información valiosa sobre el proceso de enseñanza-aprendizaje.

Los objetivos de aprendizaje pueden redactarse siguiendo alguna de las taxonomías que han sido desarrolladas por varios educadores. Los objetivos del aprendizaje por lo general están diseñados siguiendo la Taxonomía de Bloom. Esta taxonomía contempla seis categorías para clasificar el comportamiento cognitivo: conocimiento, comprensión, aplicación, análisis, síntesis y evaluación. Estas categorías están en jerarquía de la demanda intelectual del estudiante.

1.9.1 Reactivo.

Es el planteamiento de una situación que requiere solución, que propone acciones o suscita reacciones que se traducen en respuestas, cuyo grado de acierto puede ser indicador del aprendizaje obtenido.

Consta de los siguientes componentes: *la instrucción, la base y las opciones de respuesta.*

1.9.2 El reactivo de opción múltiple

Es un planteamiento que puede presentarse en forma de pregunta o de afirmación, a la cual el examinado solamente debe elegir la correcta de entre 4 o 5 opciones, sin oportunidad de escribir o expresar nada. [22]

Existen dos tipos de reactivos de opción múltiple:

- El reactivo simple y
- El multireactivo,

Y 6 principales formatos en los que se pueden presentar:

- a) Cuestionamiento directo.
- b) Canevá o de completar.
- c) Ordenamiento y selección de elementos.
- d) Relación de columnas o apareamientos.
- e) Opción alterna o falso-verdadero.

CAPÍTULO 2: ANÁLISIS DEL SISTEMA

2.1 Planteamiento del Problema

Desde hace unos años, la Secretaría de Educación Pública (SEP), aplica el Examen General de conocimientos (EXANNII) y el examen intermedio de conocimientos, a los alumnos de las Escuelas Normales que cursan el sexto y cuarto semestre de la Licenciatura en Educación Primaria, Plan de estudios 1997. Evaluación que ha estado a cargo del Centro Nacional de Evaluación (CENEVAL) y que aporta información a los sustentantes, docentes y directivos de las escuelas normales y a las autoridades educativas, estatales y federales sobre el nivel de logro alcanzado por los estudiantes, con respecto a algunos conocimientos y habilidades esenciales establecidos en el plan de estudios.

En los cuadros siguientes se puede observar los resultados obtenidos en las evaluaciones que ha aplicado el CENEVAL. En estos se aprecia que no se ha podido alcanzar la media nacional, por lo que la institución ha implementado distintas estrategias para lograrlo. Una de ellas es la aplicación de pruebas objetivas. [23]

Estas pruebas, además de resultar costosas, requieren de mucho tiempo para calificarlas y sistematizar los resultados.

. Encuesta aplicada en 2005

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	57.81

Encuesta aplicada en 2006

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	54.71

Encuesta aplicada en 2007.

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	59.12

Encuesta aplicada en 2008.

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	55.17

Encuesta aplicada en 2009. Alumnos de sexto semestre.

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	53.37

Encuesta aplicada en 2010. Alumnos de sexto semestre.

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	52.53

Encuesta aplicada en 2011. Aplicación Mayo

Nombre de la Escuela	Clave de la Escuela	Promedio global
Normal Oficial "Profr. Luis Casarrubias Ibarra"	PUEENO015	49.16

Figura 2.1 Cuadros estadísticos, resultados por niveles de competencia por escuela de la licenciatura en Educación Primaria

2.2 Estrategia de solución

Con la finalidad de ayudar a los alumnos de la Esc. Normal Oficial "Profr. Luis Casarrubias Ibarra" a aprobar los exámenes que aplica la SEP a través del CENEVAL y de apoyar para que las características del egresado respondan a las exigencias del perfil de egreso de la Licenciatura en Educación Primaria, se requiere diseñar un sistema WEB que permita a la Institución Educativa la aplicación de pruebas objetivas y la sistematización de los resultados, permitiéndoles obtener parámetros y referencias para guiar los esfuerzos de mejoramiento educativo y la toma de decisiones de manera rápida y oportuna.

Se propone diseñar una aplicación WEB que genere pruebas objetivas, que permita a los docentes, automatizar los procesos de registro de usuarios, la creación de un banco de reactivos, generar y aplicar de forma aleatoria pruebas objetivas, visualizar reporte de calificaciones con gráficas para comparar los resultados entre los estudiantes de un grupo.

El nivel cognoscitivo de las pruebas que se podrán diseñar, será integrando elementos de la Taxonomía de los Objetivos Educativos propuesta por Benjamín S. Bloom. Se aplicará lo referente a pruebas objetivas, conocidas también como exámenes de opción múltiple o de respuesta estructurada.

Estas pruebas se conformarán con preguntas seleccionadas aleatoriamente de un banco de reactivos. Cada pregunta constará con cuatro posibles respuestas. El estudiante podrá elegir solamente la opción que incluya la respuesta correcta. Dichas preguntas no requerirán que el alumno escriba o redacte la respuesta ni desarrolle temas abiertos.

El tipo de reactivo de opción múltiple será de tipo simple. El reactivo simple es un cuestionamiento o una afirmación que requiere ser completada con la opción correcta; implica un solo reactivo, es decir no parte de un planteamiento general ni desprende otros reactivos de éste.

Los formatos del reactivo que se podrán ingresar serán:

- Cuestionamiento directo
- Tipo canevá o de completar
- Ordenamiento y selección de elementos

Se sugiere que las preguntas que se ingresen sean desde el nivel más simple que es recordar, hasta el de resolver problemas. Esta taxonomía es de las más aplicadas en instrumentos de medición de conocimientos y objetivos de aprendizaje.

Para la evolución por etapas del Sistema, se desarrollará una metodología propia de elaboración de exámenes por computadora, que consiste en los siguientes pasos:

- Desarrollo del prototipo del sistema de exámenes en línea (Registro, Evaluación, Elaboración de Reportes, Récords de sustentantes).
- Elaboración de reactivos de los temas de estudio de algunas asignaturas que conforman el plan curricular de la Licenciatura en Educación Primaria 1997
- Construcción del examen de algunos contenidos temáticos para su prueba en línea.
- Prueba de los reactivos y depuración de los mismos.
- Pruebas de implementación del Sistema.

El modelo que se utilizará para los procesos de software será el de cascada, el cual, considera las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución, y los representa como fases separadas del proceso, tales como la especificación de requerimientos, el diseño de software, la implementación, las pruebas, etcétera.

El sistema se implementará utilizando el lenguaje de programación dinámica de sitios web PHP. Se escogió este lenguaje por varias razones, entre las que cabe destacar:

- Permite gestionar eficientemente cualquier elemento de una página web.
- Se comunica perfectamente con HTML y JavaScript, permitiendo alcanzar un elevado nivel de integración con el navegador.
- Gestionar con eficacia bases de datos.
- Es un lenguaje de programación totalmente libre (open source).

Y con el gestor de bases de datos en internet MySQL que es también open source y además sensiblemente más rápido y eficiente que la mayoría de los que hay en el mercado, permite una gran versatilidad a la hora de almacenar y gestionar todo tipo de datos mediante un lenguaje específico conocido como SQL que se integra perfectamente con PHP.

2.3 Especificación de requerimientos

R.0. Requisitos generales

R.0.1 Permitir el ingreso al sistema sólo a usuarios registrados a través de su nombre de usuario y contraseña.

R.0.2 Contar con tres tipos de usuarios: Usuario Administrador, Usuario Docente y Usuario Alumno.

R.0.3 Ingresar Usuarios.

R.0.4 Edición de usuarios

R.0.5 Visualizar usuarios dados de alta en el sistema

R.0.6 Ingresar preguntas al banco de reactivos.

R.0.7 Editar preguntas del banco de reactivos

R.0.8 Generar y aplicar de forma aleatoria pruebas objetivas conformadas por preguntas de uno o varios temas y de una o varias asignaturas. Cada pregunta deberá tener cuatro posibles respuestas

R.0.9 Asignar exámenes a alumnos, por grupo o individualmente.

R.0.10 Generar reportes de los resultados de exámenes aplicados.

R.1 Requisitos Usuario Administrador

R.1.1. Cada Administrador deberán contar con un identificador único.

R.1.2 Administrar Usuarios

R.1.2.1 Podrá dar de alta a Usuarios Administradores, Usuarios Docentes y Usuarios Alumnos.

R.1.2.2 Podrá eliminar a Usuarios Administradores, Usuarios Docentes y Usuarios Alumnos.

R.1.2.2 Podrá modificar datos de Usuarios Administradores, Usuarios Docentes y Usuarios Alumnos.

R.1.3 Administrar Banco de Reactivos

R.1.3.1 Podrá generar reactivos de cualquier asignatura.

R.1.3.2 Podrá eliminar reactivos de cualquier asignatura.

R.1.3.3 Podrá modificar reactivos de cualquier asignatura.

R.1.4 Administrar Pruebas

R.1.4.1 Podrá generar pruebas de una o varias asignaturas.

R.1.4.2 Podrá especificar a qué alumnos se les aplicará el examen

R.1.4.3 Podrán eliminar exámenes generados por los Docentes Usuarios

R.1.5 Reportes

R.1.5.1 Podrán generar reportes de calificaciones de cualquier alumno

R.2 Requisitos Usuario Docente

R.2.1. Cada docente deberán contar con un identificador único

R.2.2 Administrar usuarios

R.2.2.1 Únicamente podrá ingresar alumnos

R.2.2.2 Podrá editar los datos de los alumnos

R.2.3. Generar banco de reactivos

R.2.3.1 Podrá ingresar preguntas al banco de reactivos

R.2.3.2 Podrá modificar o eliminar sólo los reactivos que haya ingresado.

R.2.4. Generar exámenes

R.2.4.1. Únicamente podrá generar pruebas de las asignaturas que el Administrador le asignó.

R.2.4.2 Podrá eliminar pruebas que hayan generado

R.2.5 Asigna Pruebas

R.2.5.1 Podrá asignar pruebas que el haya generado y sólo a los alumnos de su asignatura.

R.2.6. Generar Reportes.

R.2.6.1. Generará reportes de calificaciones únicamente de los alumnos que hayan realizado sus pruebas.

R.3 Requisitos Alumnos

R.3.1 Cada alumno deberá contar con un identificador único.

R.3.2 Únicamente podrá ingresar al sistema los alumnos que hayan sido dados de alta por el Usuario Administrador.

R.3.3 Evaluación

R.3.3.1 Podrá contestar únicamente la evaluación que les asigne el Usuario Administrador o el Usuario Docente.

R.3.3.2 Podrá consultar los resultados obtenidos en sus evaluaciones.

2.4 Diagramas de Casos de Uso

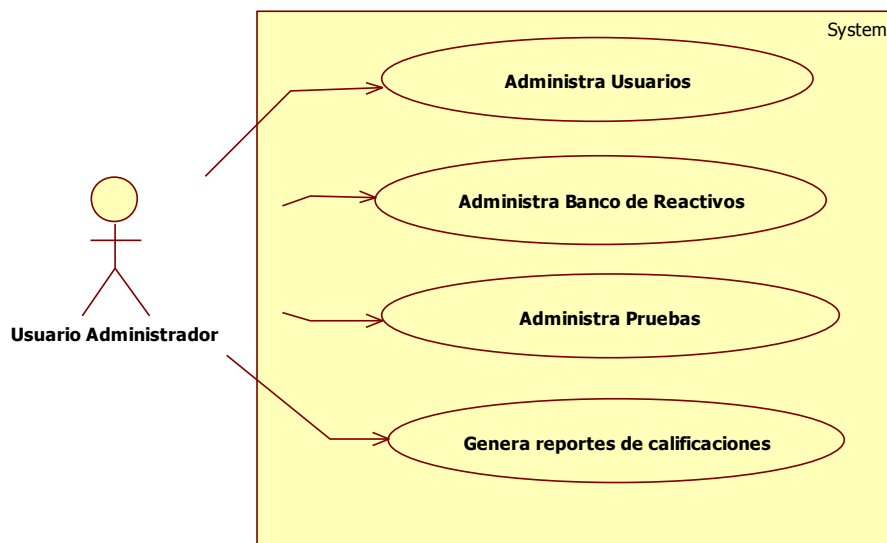


Figura. 2.2 Diagrama de caso de uso: usuario Administrador

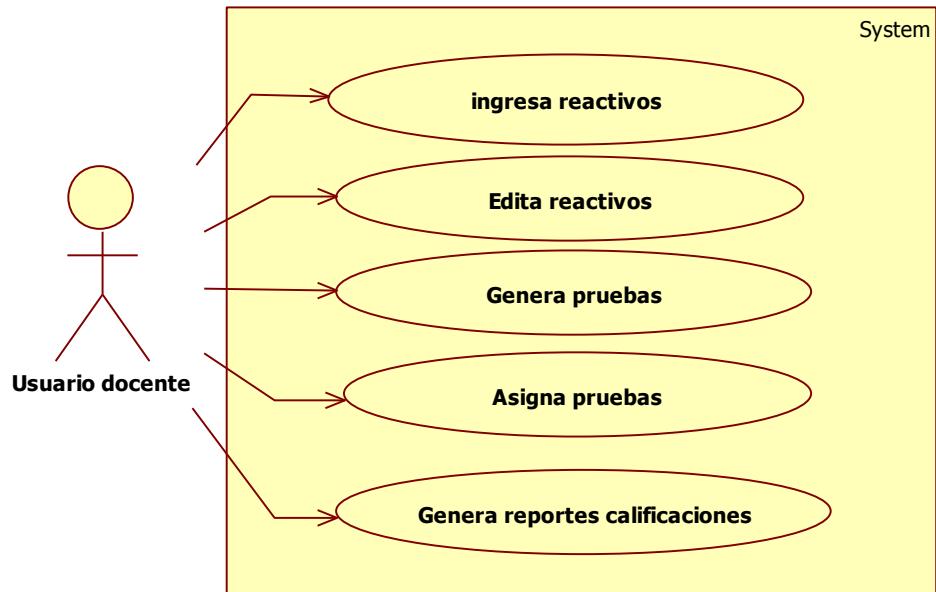


Figura. 2.3 Diagrama de caso de uso: usuario Docente

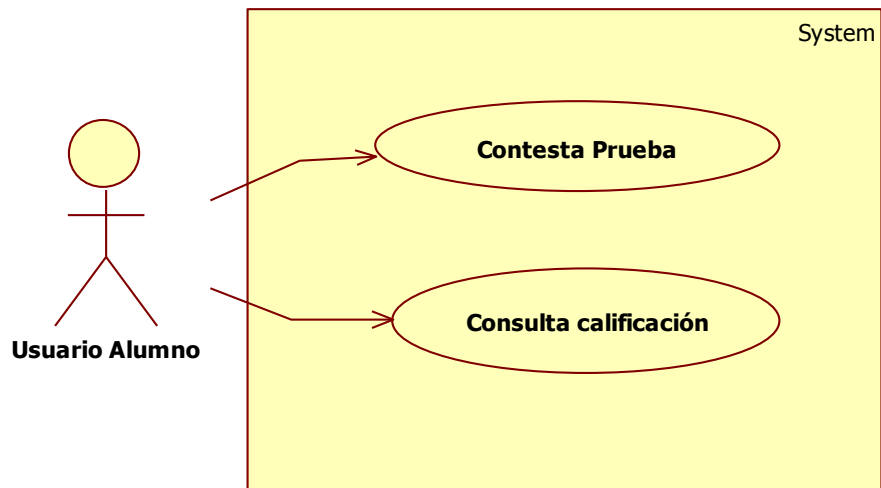


Figura. 2.4 Diagrama de caso de uso: usuario Alumno

2.5 Secuencias de acciones posibles para los casos de uso.

Docente Administrador

1. Secuencia de acciones para un camino del caso de uso Administra Usuarios.

El Usuario Administrador ingresa, modifica o elimina del sistema a usuarios administradores, usuarios docentes y Alumnos.

2. Secuencia de acciones para un camino del caso de uso Administrar Banco de Reactivos.

Los usuarios Administradores ingresan, modifican o eliminan preguntas del banco de reactivos de cualquier asignatura.

3. Secuencia de acciones para un camino del caso de uso Administra Pruebas

El Usuario Administrador genera pruebas de cualquier asignatura, especifica a qué alumnos se les aplicará o elimina pruebas generadas por cualquier usuario.

4. Secuencia de acciones para un camino del caso de uso Genera reportes de calificaciones.

El Usuario Administrador genera reporte de calificaciones de cualquier alumno.

Usuario Docente

1. Secuencia de acciones para un camino del caso de uso Ingresa reactivos

El usuario Docente inserta preguntas al banco de reactivos de las asignaturas asignadas por el Usuario Administrador.

2. Secuencia de acciones para un camino del caso de uso Edita Reactivos

El usuario Docente modifica o elimina preguntas que ha insertado.

3. Secuencia de acciones para un camino del caso de uso Genera Pruebas

El usuario Docente genera pruebas con preguntas de uno o varios temas de las materias asignadas por el Usuario Administrador.

4. Secuencia de acciones para un camino del caso de uso Asigna Pruebas

El usuario Docente asigna a sus alumnos, la prueba que ha elaborado.

5. Secuencia de acciones para un camino del caso de uso Genera reportes de calificaciones.

El Usuario Docente genera reporte de calificaciones de sus alumnos.

Usuario Alumno

1. Secuencia de acciones para un camino del caso de uso Contesta Prueba

El usuario Alumnos contesta la evaluación asignada.

2. Secuencia de acciones para un camino del caso de uso Consulta Calificación

El usuario Alumnos consulta la calificación obtenida en la prueba.

2.6 Especificación de Casos de Uso del sistema

1. Ingreso Usuarios.

Nombre:	Ingresar Docentes
Objetivo:	Ingreso de usuarios Administradores o usuarios Docentes
Precondiciones:	Ingresar como usuario administrador
Actores	Usuario Administrador

FLUJOS DE EVENTOS

Ingresar Docentes	1 El actor selecciona la opción Ingresar Docente
	2 El sistema le muestra el formulario de alta de docentes.
	3 El actor ingresa los siguientes datos:
	I. Apellido Paterno
	II. Apellido Materno
	III. Nombre
	IV. Nombre de Usuario
	V. Contraseña
	VI. Marca la opción Administrador o Docente
	4 El actor acepta el registro mediante el botón Guardar
	5 El sistema guarda el registro y manda un mensaje

de confirmación.

Excepciones:	En el punto cuatro del flujo normal el sistema verifica si falta información obligatoria. Se muestra un mensaje indicándolo y se vuelve al paso anterior para que el actor ingrese la información faltante.
Interfaz de usuario:	Pantalla del menú de opciones

2. Edición Usuarios

Nombre:	Edición Usuarios
Objetivo:	Modificación o Alta de usuarios Administradores o usuarios Docentes
Precondiciones:	Ingresar como usuario administrador
Actores:	Usuario Administrador

FLUJOS DE EVENTOS

Modificar	1	El actor selecciona la opción Editar Docentes
	2	El sistema le muestra todos los usuarios administradores y usuarios docentes registrados
	3	El actor oprime el botón EDITAR que corresponde al usuario seleccionado
	4	El actor realiza los cambio
	5	El actor acepta los cambios mediante el botón modificar
	6	El sistema guarda los cambios y regresa a la lista de usuarios registrados

FLUJOS ALTERNOS

Eliminar	1	El actor oprime el botón ELIMINAR
	2	El sistema elimina el registro y manda un mensaje de confirmación

3 El sistema regresa a la lista de registrados

Interfaz de usuario: Pantalla del menú de opciones

3. Ingresar Alumnos

Nombre: Ingresar Alumnos
Objetivo: Ingreso de usuarios Alumnos
Precondiciones: Ingresar como usuario administrador
Actores: Usuario Administrador

FLUJOS DE EVENTOS

Ingresar Alumnos

- 1 El actor selecciona la opción Ingresar Alumnos
- 2 El sistema le muestra el formulario de alta de Alumnos
- 3 El actor ingresa los siguientes datos:
 - I. Apellido Paterno
 - II. Apellido Materno
 - III. Nombre
 - IV. Matrícula
 - V. Semestre
 - VI. Grupo
 - VII. Contraseña
- 4 El actor acepta el registro mediante el botón Guardar
- 5 El sistema guarda el registro y manda un mensaje de confirmación.

Excepciones: En el punto cuatro del flujo normal el sistema verifica si falta información obligatoria. Se muestra un mensaje indicándolo y se vuelve al paso anterior para que el actor ingrese la información faltante.

Interfaz de usuario: Pantalla del menú de opciones

4. Edición Alumnos

Nombre:	Edición Alumnos
Objetivo:	Modificación o Alta de usuarios Alumnos
Precondiciones:	Ingresar como usuario administrador
Actores	Usuario Administrador

FLUJOS DE EVENTOS

- | | |
|-----------|--|
| Modificar | <ol style="list-style-type: none">1 El actor selecciona la opción Editar Alumnos
El actor selecciona la opción de búsqueda de alumnos registrados (por matrícula o por semestre)2 El sistema le muestra los usuarios Alumnos encontrados3 El actor oprime el botón EDITAR que corresponde al usuario seleccionado4 El actor realiza los cambio5 El actor acepta los cambios mediante el botón modificar6 El sistema guarda los cambios y regresa a la opción de buscar usuarios |
|-----------|--|

FLUJOS ALTERNOS

- | | |
|----------|--|
| Eliminar | <ol style="list-style-type: none">1 El actor oprime el botón ELIMINAR2 El sistema elimina el registro y manda un mensaje de confirmación3 El sistema regresa a la opción Buscar Usuarios |
|----------|--|

Interfaz de usuario:	Pantalla del menú de opciones
----------------------	-------------------------------

5. Ingresar Reactivos

Nombre:	Ingresar Reactivos
Objetivo:	Ingresar preguntas en el banco de reactivos
Precondiciones:	Ingresar como usuario administrador o usuario Docente
Actores	Usuario Administrador

FLUJOS DE EVENTOS

Ingresar Reactivo

- 1 El actor selecciona la opción Ingresar Reactivo
 - 2 El sistema le muestra el acta de nacimiento del reactivo
 - 3 El actor ingresa los siguientes datos:
 - I. Instrucción
 - II. Base
 - III. Opción A
 - IV. Opción B
 - V. Opción C
 - VI. Opción D
 - VII. Opción correcta
 - VIII. Formato del reactivo
 - IX. Nivel Taxonómico
 - X. Justificación y Fuente
 - XI. Semestre
 - XII. Materia
 - XIII. Tema
 - XIV. Subtema
 - XV. Fecha
 - XVI. Elaborador
 - 4 El actor acepta el registro mediante el botón Guardar
 - 5 El sistema guarda el registro y manda un mensaje de confirmación.
-

Excepciones:	En el punto cuatro del flujo normal el sistema verifica si falta información obligatoria. Se muestra un mensaje indicándolo y se vuelve al paso anterior para que el actor ingrese la información faltante.
Interfaz de usuario:	Pantalla del menú de opciones

6. Edición Reactivos

Nombre:	Edición Reactivos
Objetivo:	Modificación o baja de Reactivos
Precondiciones:	Ingresar como usuario administrador
Actores	Usuario Administrador

FLUJOS DE EVENTOS

Modificar	1	El actor selecciona la opción Editar Reactivo El actor selecciona el semestre y la materia
	2	El sistema le muestra los reactivos encontrados
	3	El actor oprime el botón EDITAR que corresponde al reactivo seleccionado
	4	El actor realiza los cambio
	5	El actor acepta los cambios mediante el botón modificar
	6	El sistema guarda los cambios y manda un mensaje de confirmación El sistema regresa a la lista de reactivos encontrados

FLUJOS ALTERNOS

Eliminar	1	El actor oprime el botón ELIMIINAR
	2	El sistema elimina el registro y manda un mensaje de confirmación

- 3 El sistema regresa a la lista de reactivos encontrados

Interfaz de usuario: Pantalla del menú de opciones

7. Generar Prueba

Nombre: Generar Prueba
 Objetivo: Generar las pruebas que se van a aplicar
 Precondiciones: Ingresar como usuario docente
 Actores: Usuario Administrador, usuario Docente

FLUJOS DE EVENTOS

- | | |
|----------------|--|
| Generar prueba | <ol style="list-style-type: none"> 1 El actor selecciona la opción Generar Prueba
El sistema muestra el formulario Crea Prueba 2 El actor ingresa los siguientes datos: <ol style="list-style-type: none"> I. Nombre de la Prueba II. Semestre III. Materia IV. Tema V. Subtema VI. Num. de preguntas 3 El actor oprime el botón CREA PRUEBA 4 El actor decide si desea ingresar más preguntas de otro tema o subtema 5 El sistema regresa al menú de opciones 6 El sistema regresa a la lista de reactivos encontrados |
|----------------|--|

Excepciones: En el punto tres del flujo normal el sistema verifica si falta información obligatoria. Se muestra un mensaje indicándolo y se vuelve al paso anterior para que el actor ingrese la información faltante.

Interfaz de usuario: Pantalla del menú de opciones

8. Generar reporte de calificaciones

Nombre:	Genera reporte de Calificaciones
Objetivo:	Consultar las calificaciones obtenidas por los alumnos
Precondiciones:	Ingresar como usuario administrador o usuario docente
Actores	Usuario Administrador y usuario docente

FLUJOS DE EVENTOS

Consultar Calificaciones	1	El actor selecciona la opción consultar calificaciones
	2	El actor selecciona la prueba de la que desea consultar las calificaciones
	3	El sistema muestra el reporte de calificaciones

FLUJOS ALTERNOS

Ver Ficha	1	El actor oprime el botón FICHA que corresponde al alumno seleccionado
	2	El sistema muestra el reporte de la prueba que contestó el alumno
Ver Gráfica	1	El actor oprime el botón GRÁFICA
	2	El sistema muestra una gráfica de resultados de la prueba

Interfaz de usuario:	Pantalla del menú de opciones
----------------------	-------------------------------

9. Contesta prueba

Nombre:	Contesta prueba
Objetivo:	Contestar la prueba asignada por el docente
Precondiciones:	Ingresar como usuario alumno
Actores	Usuario alumno

FLUJOS DE EVENTOS

- | | | |
|-----------------|---|---|
| Contesta prueba | 1 | El actor selecciona la prueba que desea contestar |
| | 2 | El sistema le muestra las preguntas de la prueba seleccionada |
| | 3 | El actor oprime el botón ENVIAR al terminar de contestar la prueba |
| | 4 | El sistema guarda la información enviada y muestra la calificación que obtuvo el alumno |
| | 5 | El sistema muestra la pantalla de ingreso al sistema |

FLUJOS ALTERNOS

- | | | |
|-------------|---|---|
| Ver Ficha | 1 | El actor oprime el botón FICHA que corresponde al alumno seleccionado |
| | 2 | El sistema muestra el reporte de la prueba que contestó el alumno |
| Ver Gráfica | 1 | El actor oprime el botón GRÁFICA |
| | 2 | El sistema muestra una gráfica de resultados de la prueba |

Excepciones:	En el punto tres del flujo normal el sistema verifica si el alumno ha contestado todas las preguntas. Se muestra un mensaje indicándolo y se vuelve al paso anterior para que el actor ingrese la información faltante.
--------------	---

Interfaz de usuario:	Pantalla del menú de opciones
----------------------	-------------------------------

2.7 Escenarios del Sistema

Los escenarios son casos específicos de los casos de uso.

Escenario1. Alta de Docentes

Omar Cardoso ingresa al sistema como usuario administrador, en la pantalla con el menú de opciones selecciona la opción “Ingresar Docentes”, el sistema le muestra una pantalla con

los datos que debe ingresar, por ejemplo: *Apellido paterno: 'García', apellido materno: 'Aguilar', nombre: 'Adolfo', usuario: 'agarcia', contraseña: 'garcia123' y privilegio: 'docente'*; llena el formulario con los datos solicitados. A continuación oprime el botón GUARDAR y si el sistema detecta que omitió algún dato lo regresa a la pantalla de captura del formulario. El usuario completa correctamente el formulario, oprime nuevamente el botón GUARDAR y el sistema guarda el registro.

Escenario 2. Edición de Docentes

Omar Cardoso ingresa al sistema como usuario administrador, en la pantalla con el menú de opciones selecciona la opción “Editar Docentes”, el sistema le muestra una pantalla con el listado de docentes dados de alta. El usuario oprime el botón EDITAR del docente seleccionado. El sistema le muestra un formulario con los datos del docente seleccionado. El usuario modifica los datos y oprime el botón GUARDAR. Si el sistema detecta que omitió algún dato lo regresa a la pantalla de captura del formulario. El usuario completa correctamente el formulario, oprime nuevamente el botón GUARDAR y el sistema modifica el registro.

Escenario 3. Alta de Alumnos

Omar Cardoso ingresa al sistema como usuario administrador, en la pantalla con el menú de opciones selecciona la opción “Ingresar Alumnos”, el sistema le muestra una pantalla con la ficha que debe ingresar, por ejemplo: *Apellido: paterno 'Aguilar', apellido materno: 'Zacatenco', nombre: 'Teresa', matrícula: '212010002', semestre: 'primero', grupo: 'A' y contraseña: 'taguilar'*; llena el formulario con los datos solicitados. A continuación oprime el botón GUARDAR y si el sistema detecta que omitió algún dato le muestra un mensaje de error y le muestra nuevamente el formulario. El usuario completa correctamente el formulario, oprime nuevamente el botón GUARDAR y el sistema guarda el registro.

Escenario 4. Edición de Alumnos

Omar Cardoso ingresa al sistema como usuario administrador, en la pantalla con el menú de opciones selecciona la opción “Editar Alumnos”, el sistema le muestra una pantalla con la opción de buscar por Matrícula y buscar por semestre. Si el usuario selecciona “buscar por matrícula”, ingresa la matrícula, oprime el botón buscar y el sistema busca el registro. Si existe la matrícula se muestra la ficha con los datos del alumno.

Si el usuario selecciona “buscar por semestre”. Selecciona el semestre y el grupo. El sistema le muestra un listado con los alumnos registrados; el usuario oprime el botón EDITAR del docente seleccionado y el sistema le muestra un formulario con los datos del docente seleccionado. El usuario modifica los datos y oprime el botón GUARDAR. Si el sistema detecta que omitió algún dato, presenta un mensaje de error y muestra nuevamente el formulario. El usuario completa correctamente el formulario, oprime nuevamente el botón GUARDAR y el sistema modifica el registro.

Escenario 5. Ingreso de reactivos

Omar Cardoso ingresa al sistema como usuario administrador, en la pantalla con el menú de opciones selecciona la opción “Ingresar Reactivos”. El sistema le muestra un formulario -que es el acta de nacimiento del reactivo- con los siguientes campos: Instrucción, base, respuesta A, respuesta B, respuesta C, respuesta D, respuesta correcta, Formato, nivel taxonómico, justificación y fuente, semestre, materia, tema, subtema, fecha, y elaborador. El usuario llena el formulario con los datos solicitados. A continuación oprime el botón GUARDAR y si el sistema detecta que omitió algún dato le muestra un mensaje de error y le muestra nuevamente el formulario. El usuario completa correctamente el formulario, oprime nuevamente el botón GUARDAR y el sistema guarda el registro.

Escenario 6. Edición de reactivos

Omar Cardoso ingresa al sistema como usuario administrador, en la pantalla “menú de opciones” selecciona la opción “Editar reactivos”. El sistema le muestra una pantalla donde debe seleccionar el docente que elaboró el reactivo y la materia. El usuario oprime el botón BUSCAR. El sistema le muestra un listado con los reactivos que corresponden a la materia y al docente que los elaboró. El usuario oprime el botón EDITAR del reactivo seleccionado y el sistema le muestra el acta de nacimiento del reactivo seleccionado. El usuario modifica los datos y oprime el botón GUARDAR. Si el sistema detecta que omitió algún dato le muestra un mensaje de error y muestra nuevamente el acta de nacimiento. El usuario completa correctamente el formulario, oprime nuevamente el botón GUARDAR y el sistema modifica el registro.

Escenario 7. Generar pruebas

Omar Cardoso ingresa al sistema como usuario Docente, en la pantalla “menú de opciones” selecciona la opción “Generar Prueba”. El sistema le muestra una pantalla con el formulario

donde debe seleccionar el docente que elaboró el reactivo y la materia. El usuario oprime el botón BUSCAR. El sistema le muestra un formulario de la información que debe ingresar: nombre del examen, semestre, materia, tema, subtema y el número de preguntas que se incluirán del subtema seleccionado. El usuario oprime el botón CREAR PRUEBA. Si el sistema detecta que omitió algún dato, le muestra un mensaje de error y muestra nuevamente el formulario. El usuario completa correctamente el formulario, oprime nuevamente el botón CREAR PRUEBA y el sistema crea un registro con los datos de la prueba generada.

Escenario 8. Asignar de pruebas

Omar Cardoso ingresa al sistema como usuario Docente, en la pantalla “menú de opciones” selecciona la opción “Asignar Prueba”. El sistema le muestra una pantalla donde debe seleccionar el semestre. El sistema le muestra un listado con los registros encontrados. El usuario marca los alumnos a los que les asignará la prueba y posteriormente oprime el botón ASIGNAR

.

Escenario 9. Reporte de calificaciones

Omar Cardoso ingresa al sistema como usuario Docente, en la pantalla “menú de opciones” selecciona la opción Reporte de Calificaciones. El sistema muestra una pantalla donde le pide que seleccione la prueba de la cual desea ver las calificaciones. El sistema muestra las calificaciones de los alumnos que realizaron la prueba.

Escenario 10. Resolución de pruebas

La alumna Teresa García Zacatenco ingresa con su matrícula y contraseña. El sistema le pide que seleccione la prueba que desea contestar. A continuación le muestra las preguntas de la prueba seleccionada. La alumna selecciona una opción de cada pregunta y oprime el botón ENVIAR. Si el sistema detecta que no contestó todas las preguntas, le muestra un mensaje de error y le muestra nuevamente la prueba. La alumna contesta todas la preguntas, oprime nuevamente el botón ENVIAR y el sistema le muestra la calificación que obtuvo.

2.8 Diagramas de Secuencia del Sistema

Se muestran los diagramas de secuencia, que son generados a partir de los casos de uso descritos en el capítulo dos.

2.8.1 Diagrama de Secuencia Ingreso Docentes

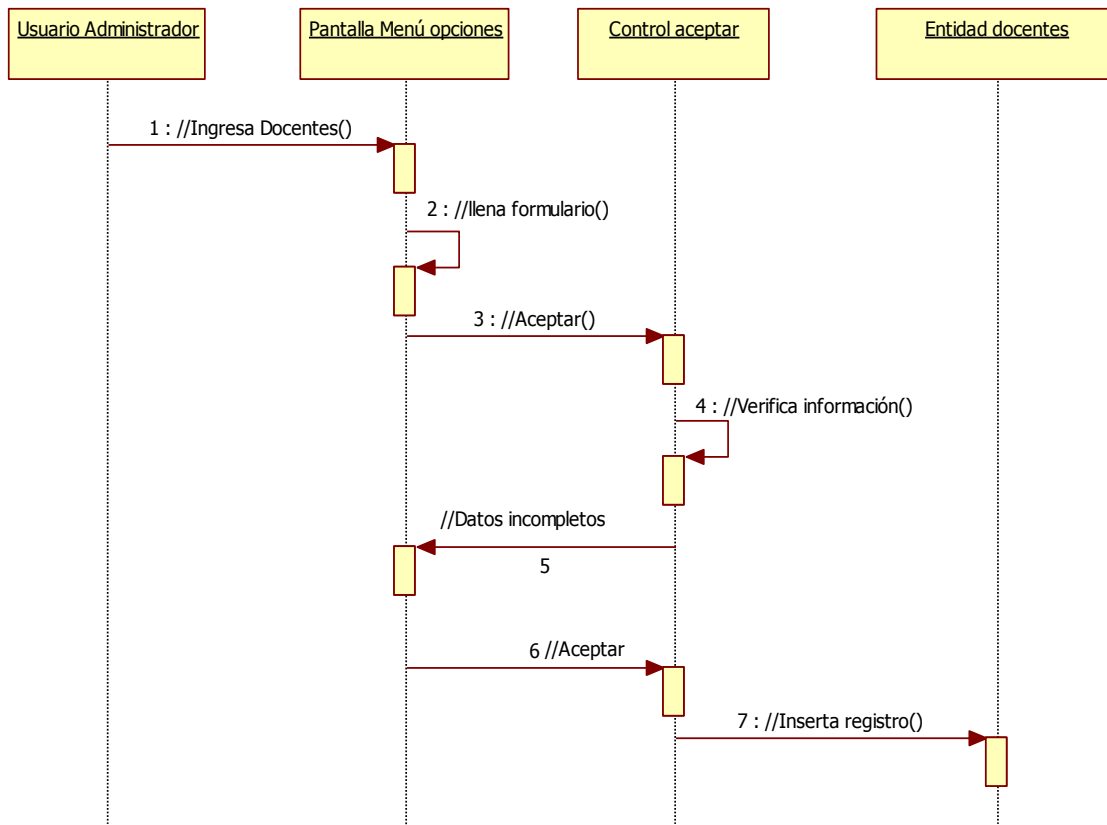


Figura 2.5 Diagrama de Secuencia Ingreso Docentes

2.8.2 Diagrama de Secuencia Editar Docentes

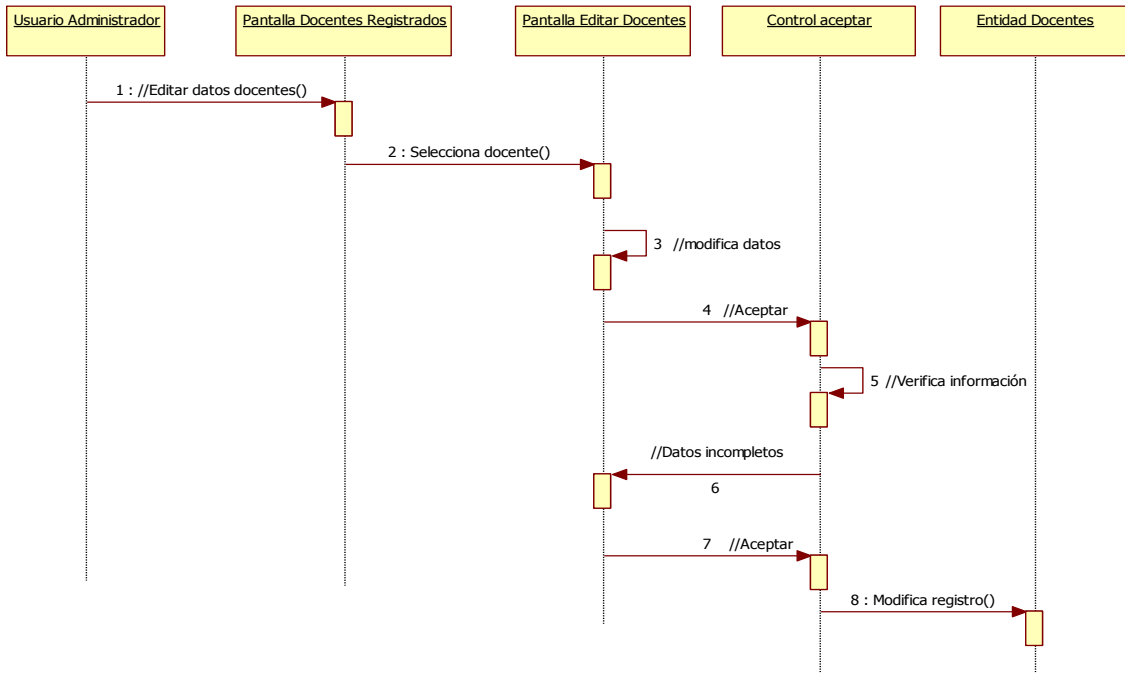


Figura 2.6 Diagrama de Secuencia Editar Docentes

2.8.3 Diagrama de Secuencia Insertar Alumnos

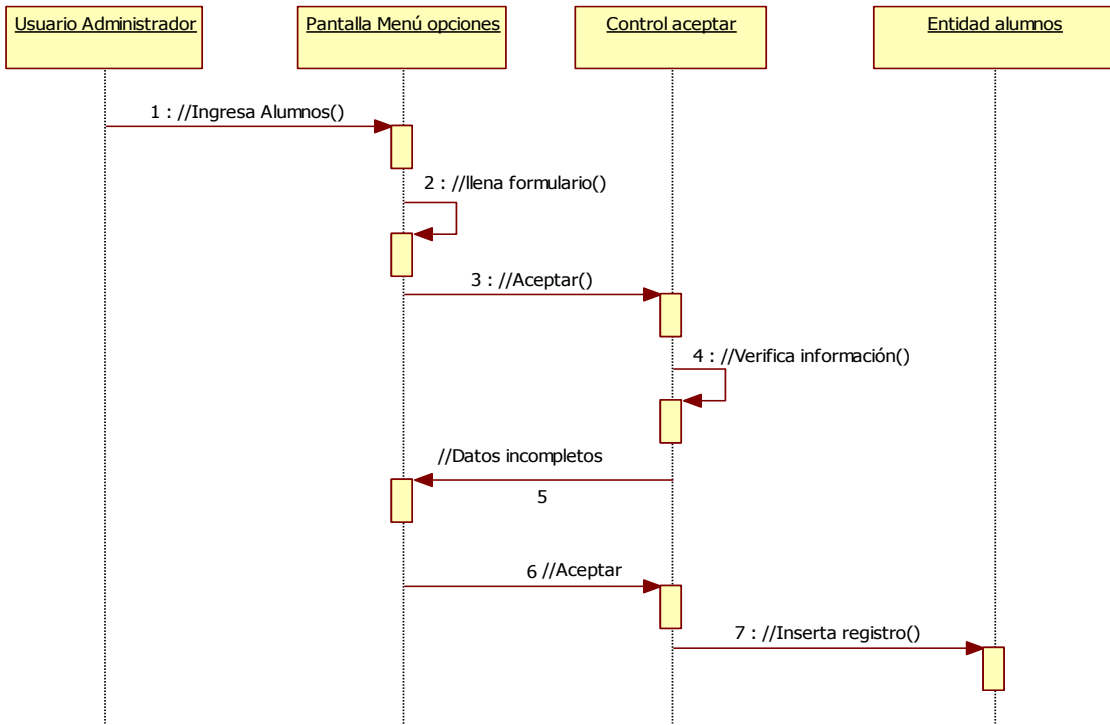


Figura 2.7 Diagrama de Secuencia Ingresa Alumnos

2.8.4 Diagrama de Secuencia Editar Alumnos

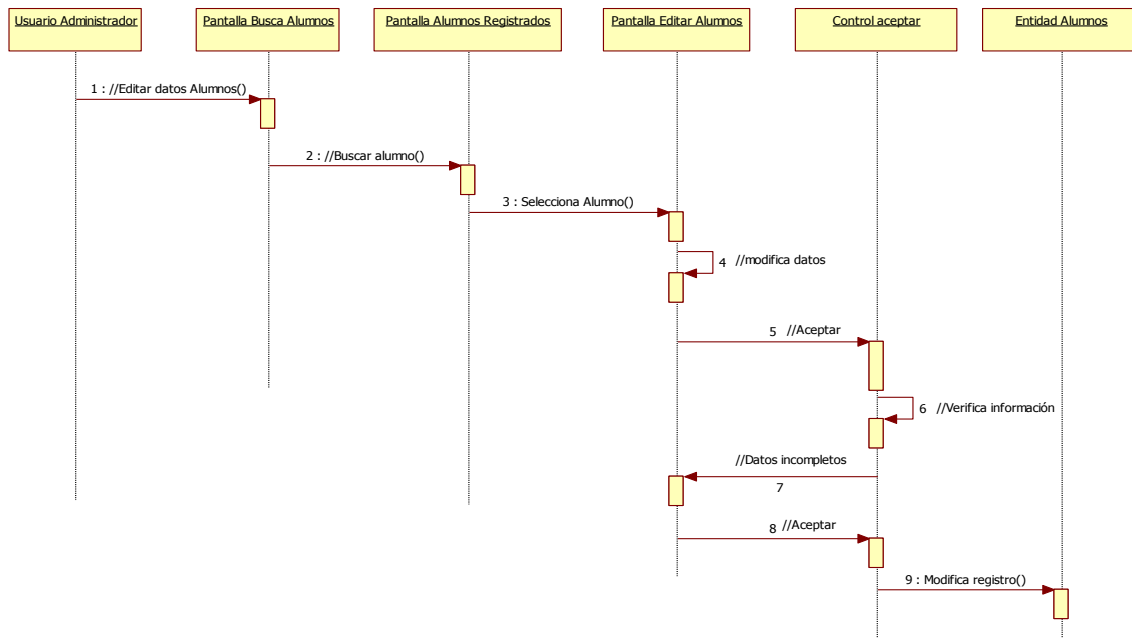


Figura 2.8 Diagrama de Secuencia Editar Alumnos

2.8.5 Diagrama de Secuencia Ingresar Reactivos

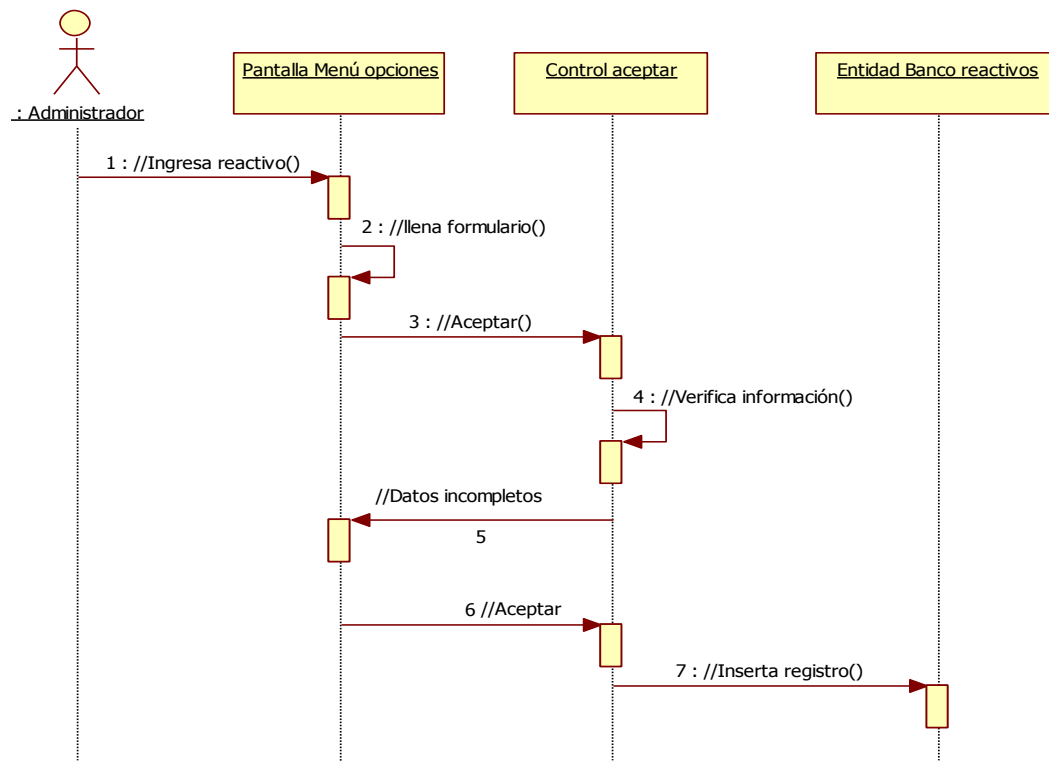


Figura 2.9 Diagrama de Secuencia Ingresar Reactivos

2.8.6 Diagrama de Secuencia Editar Reactivos

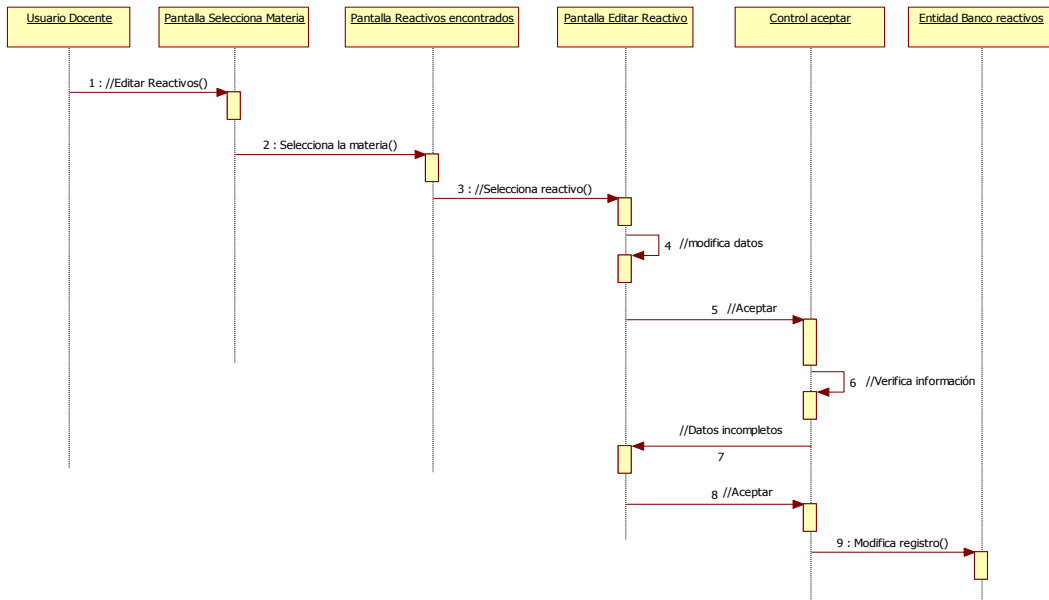


Figura 2.10 Diagrama de Secuencia Editar Reactivos

2.8.7 Diagrama de Secuencia Genera Prueba

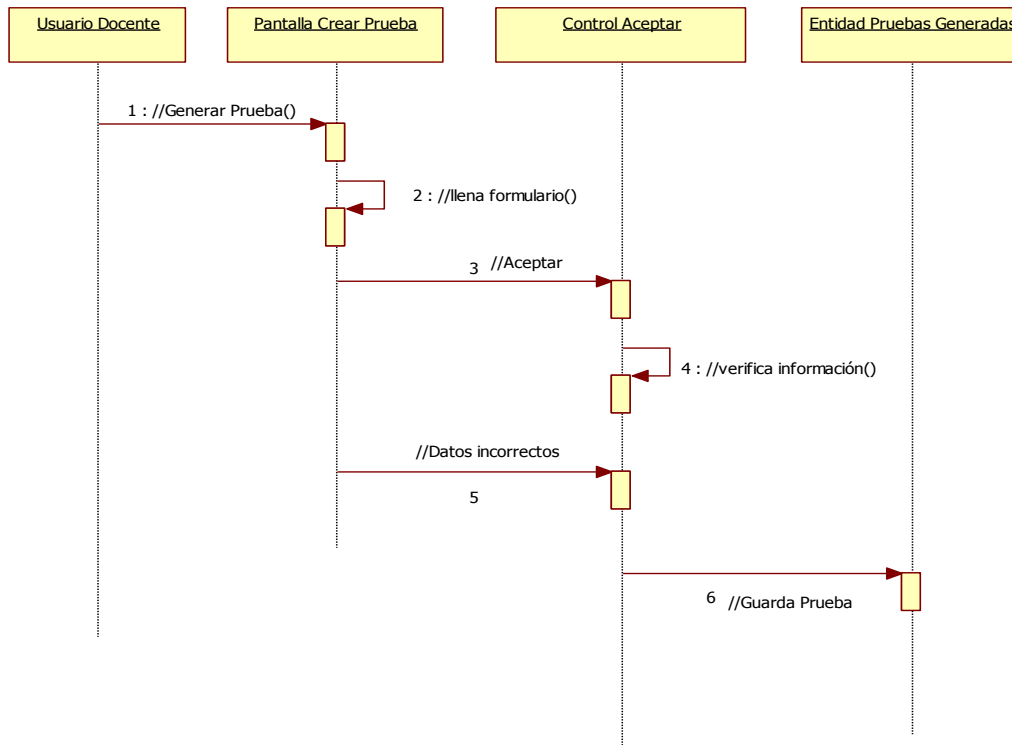


Figura 2.11 Diagrama de Secuencia Generar Prueba

2.8.8 Diagrama de Secuencia Asigna Prueba

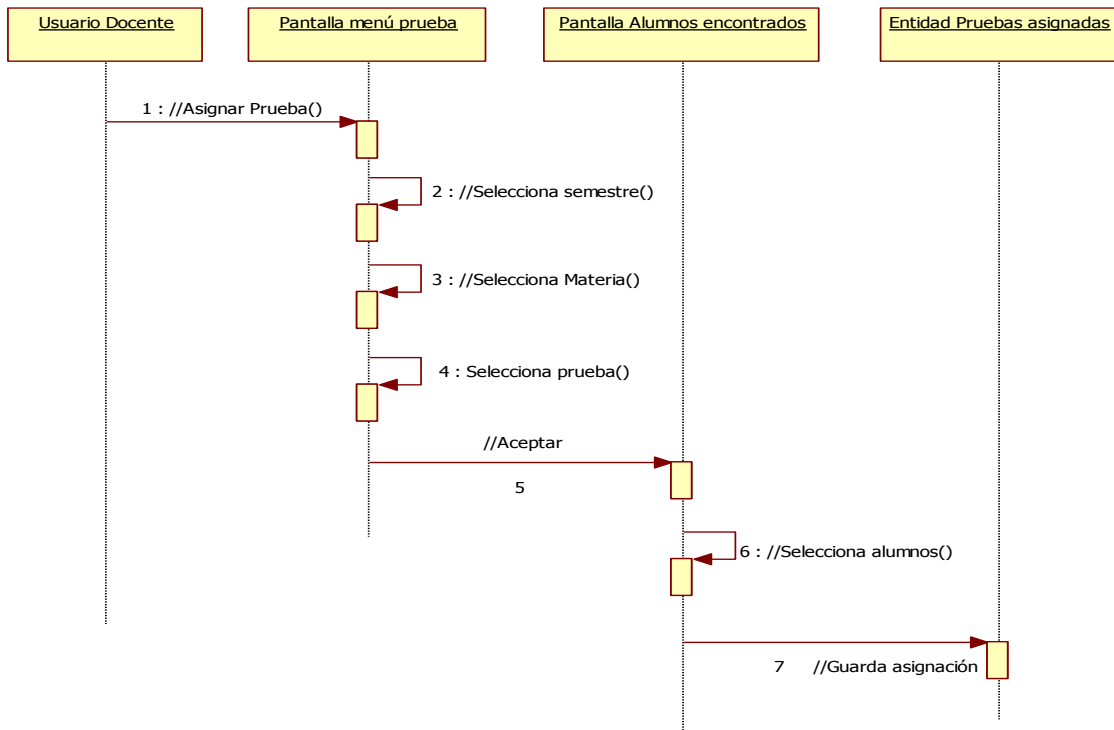


Figura 2.12 Diagrama de Secuencia Asigna Prueba

2.8.9 Diagrama de Secuencia Contesta Prueba

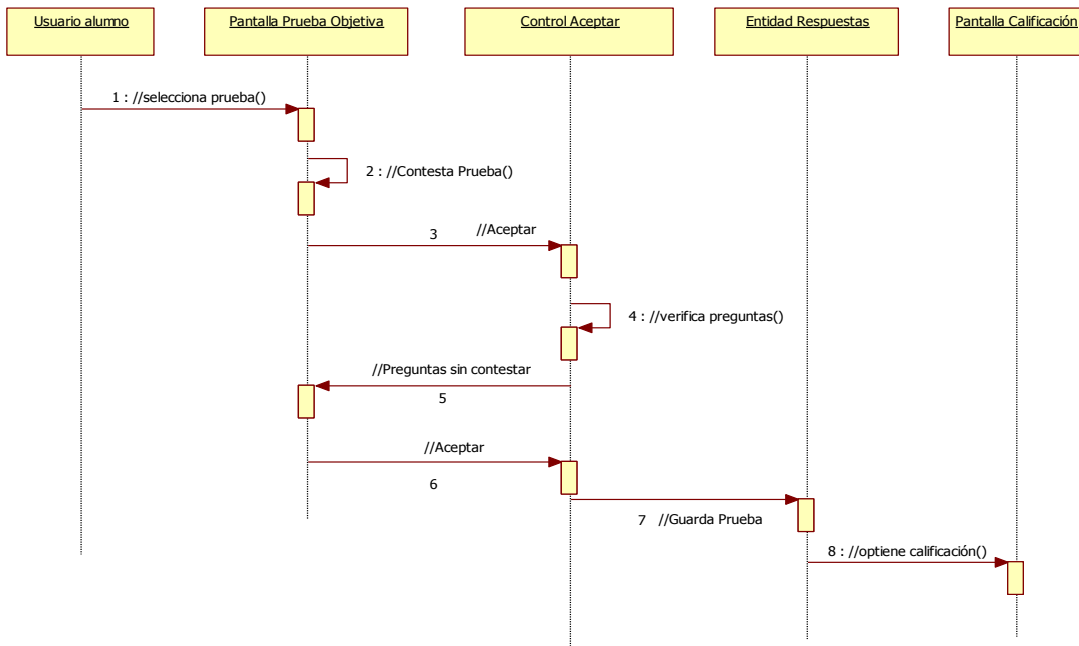


Figura 2.13 Diagrama de Secuencia Contesta Prueba

2.8.10 Diagrama de Secuencia Reporte de Calificaciones

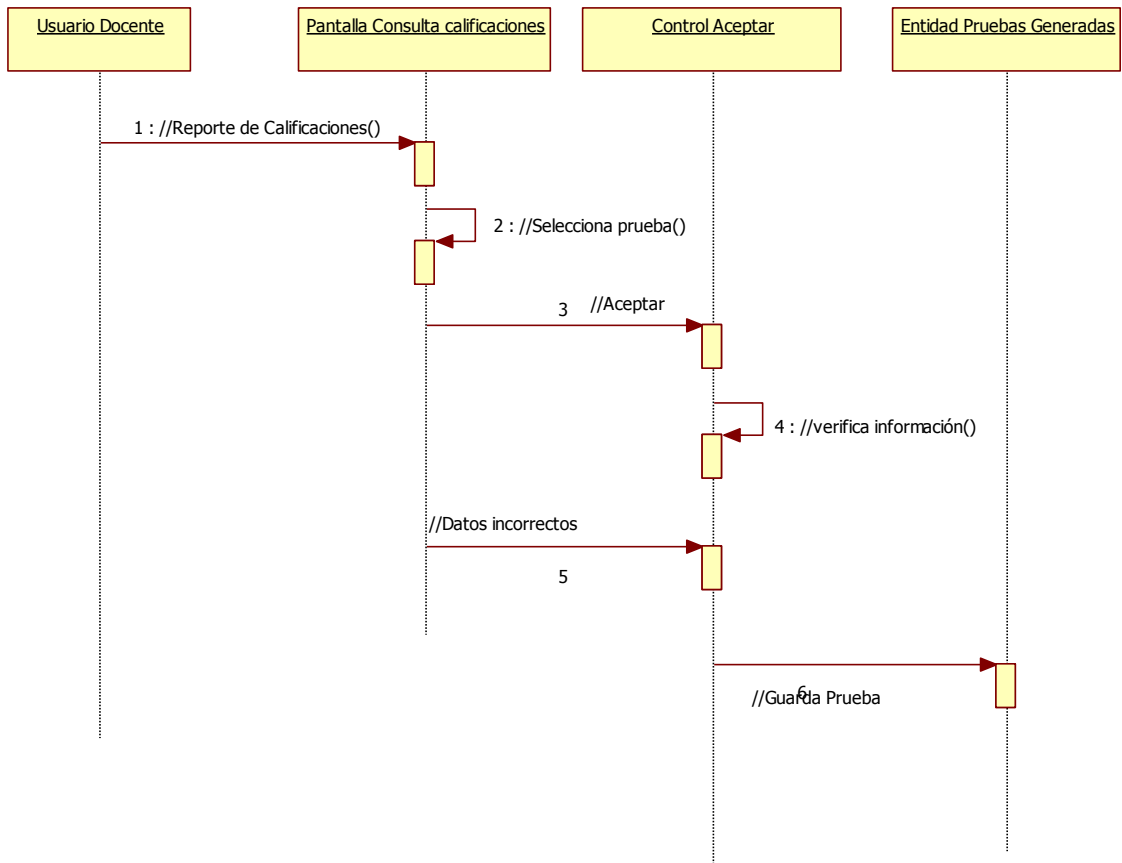


Figura 2.14 Diagrama de Secuencia Reporte de Calificaciones

2.9 Diagramas de Actividades del Sistema

Un diagrama de actividades representa los flujos de trabajo paso a paso, de negocio y operacionales de los componentes en un sistema. Un Diagrama de Actividades muestra el flujo de control general.

2.9.1 Diagrama de actividades Ingresar Docentes

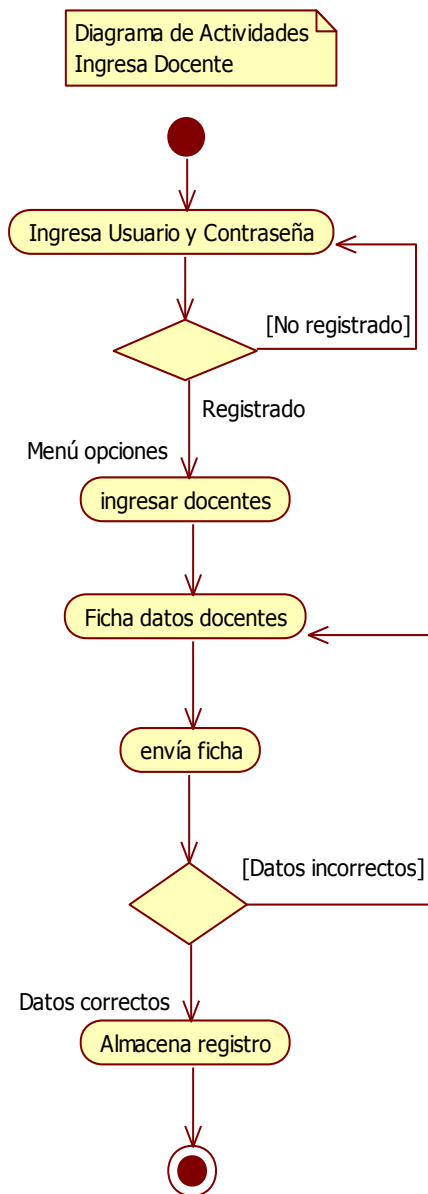


Figura 2.15 Diagrama de actividades Ingresar Docentes

2.9.2 Diagrama de actividades Modificar Docentes

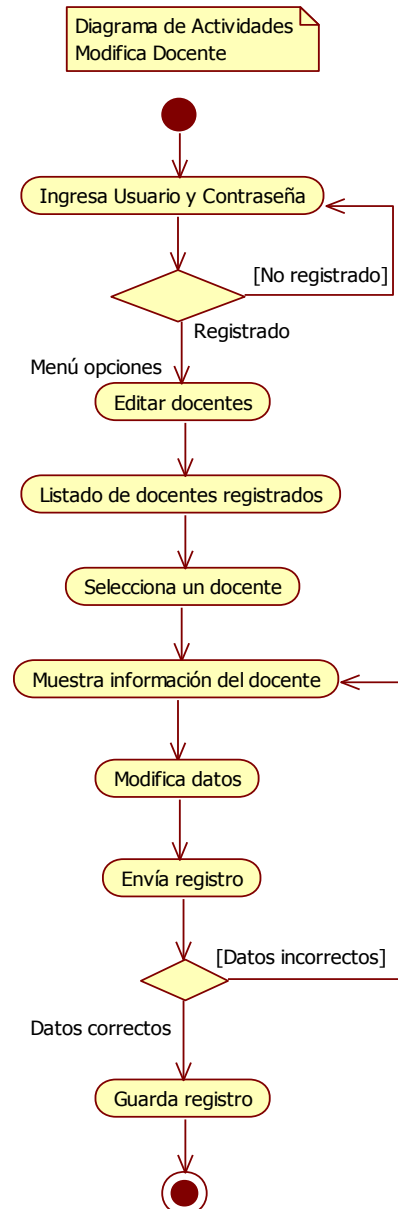


Figura 2.16 Diagrama de actividades Modificar Docentes

2.9.3 Diagrama de actividades Elimina Docentes

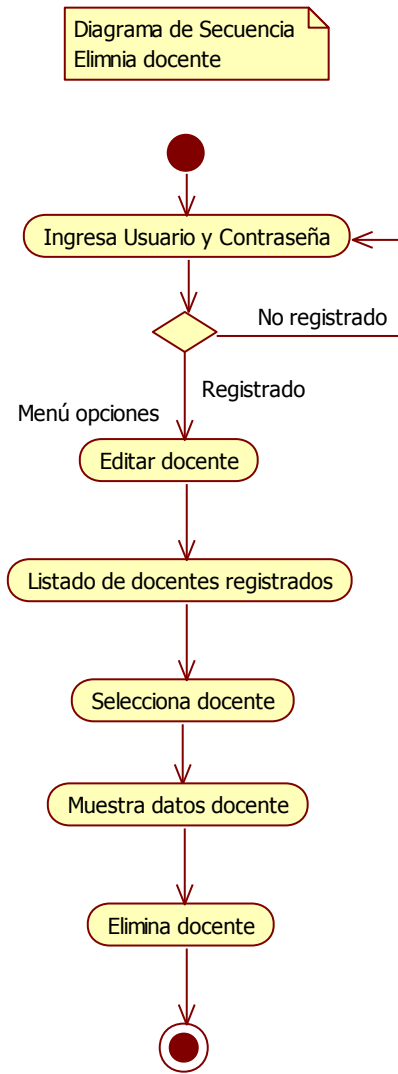


Figura 2.17 Diagrama de actividades Elimina Docentes

2.9.4 Diagrama de actividades Ingresar Alumnos

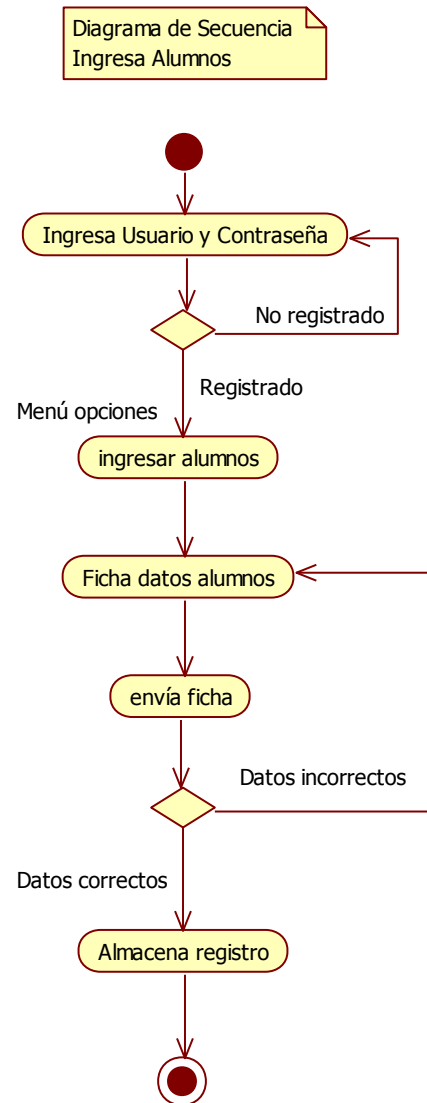


Figura 2.18 Diagrama de actividades Elimina Docentes

2.9.5 Diagrama de actividades Modifica Alumnos

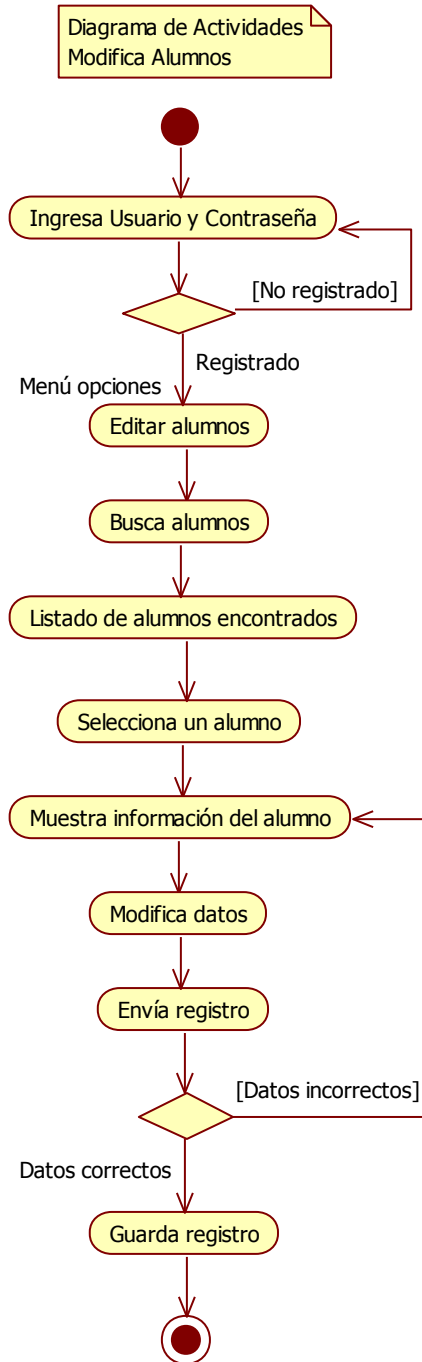


Figura 2.19 Diagrama de actividades Modifica Alumnos

2.9.6 Diagrama de actividades Elimina Alumnos

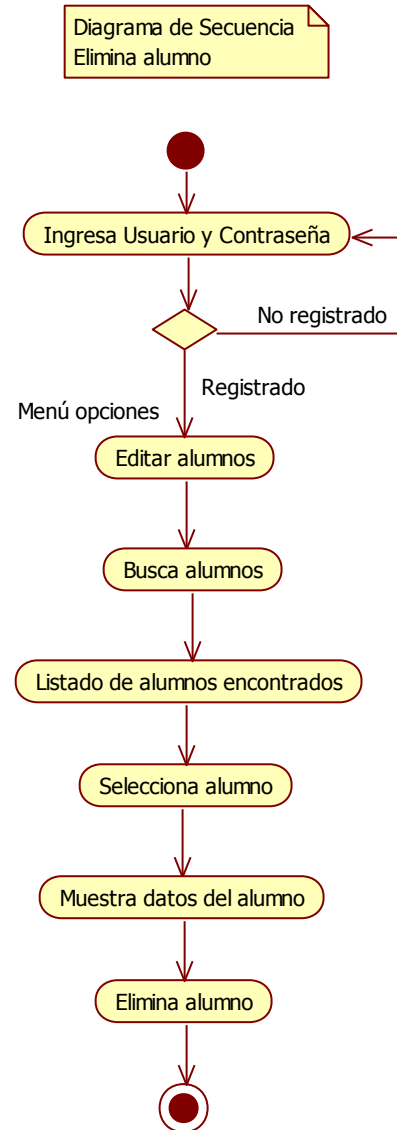


Figura 2.20 Diagrama de actividades Elimina Alumnos

2.9.7 Diagrama de actividades Ingresar Reactivo

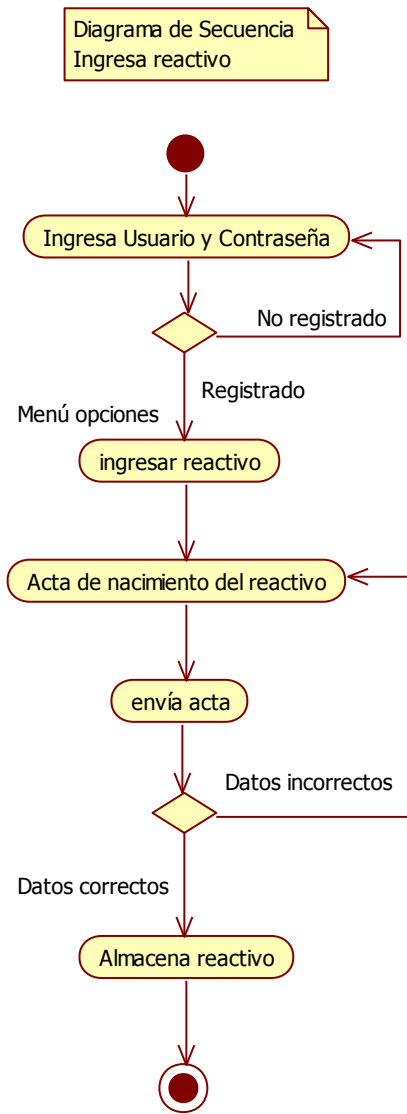


Figura 2.21 Diagrama de actividades Ingresar Reactivo

2.9.8 Diagrama de actividades Modifica Reactivo

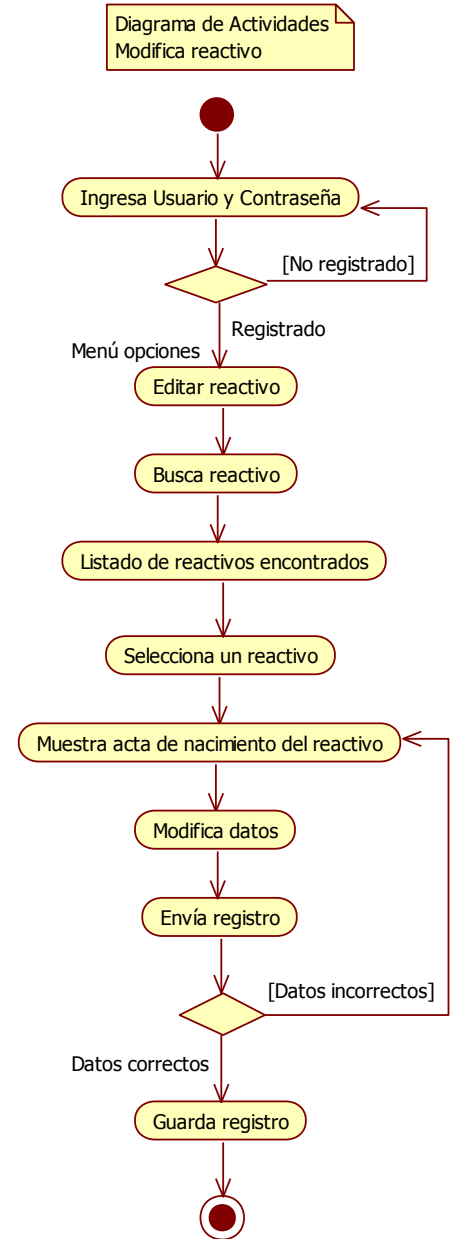


Figura 2.22 Diagrama de actividades Modifica Reactivo

2.9.9 Diagrama de actividades Elimina Reactivo

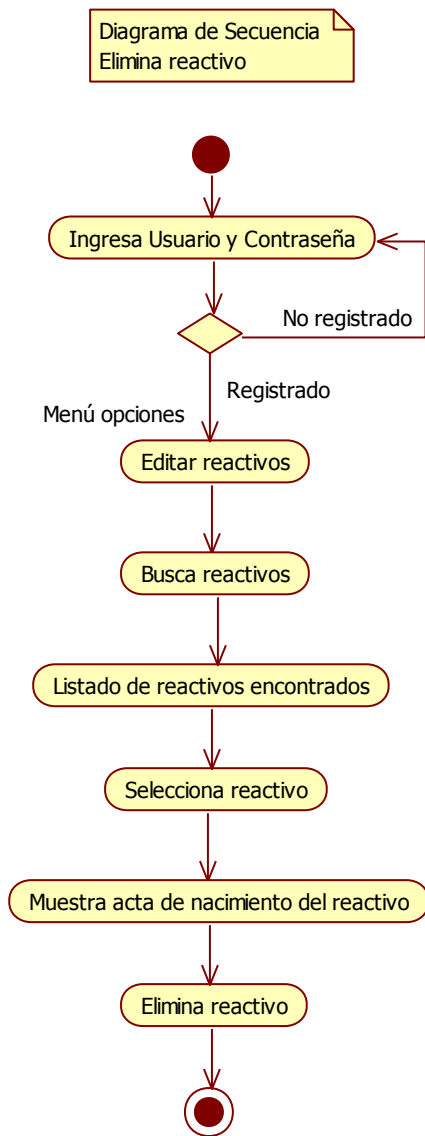


Figura 2.23 Diagrama de actividades Elimina Reactivo

2.9.10 Diagrama de actividades Visualiza Docentes

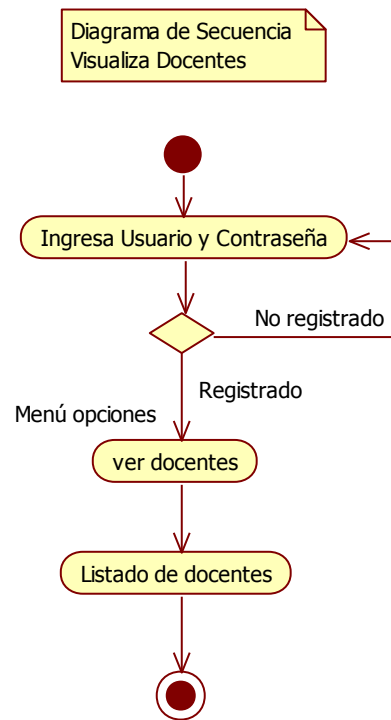


Figura 2.24 Diagrama de actividades Visualiza Docentes

2.9.11 Diagrama de actividades Visualiza Alumnos

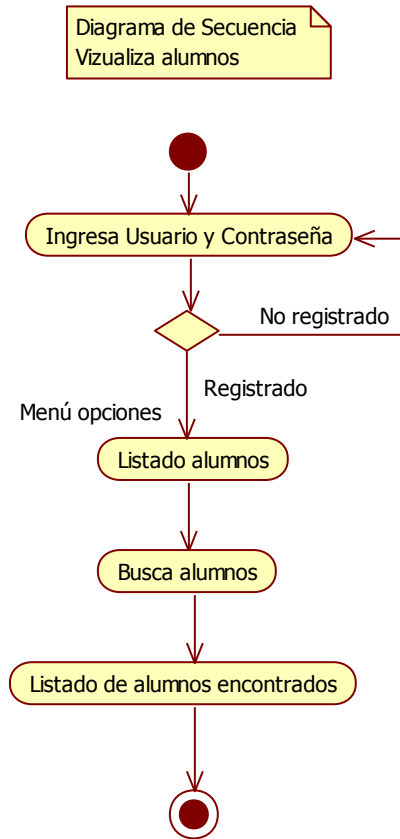


Figura 2.25 Diagrama de actividades
Visualiza Alumnos

2.9.12 Diagrama de actividades Visualiza Reactivos

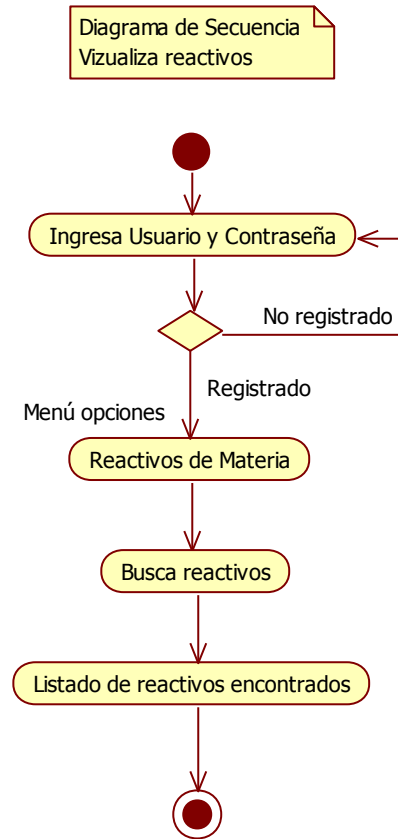


Figura 2.26 Diagrama de actividades
Visualiza Reactivos

2.9.13 Diagrama de actividades Genera Prueba

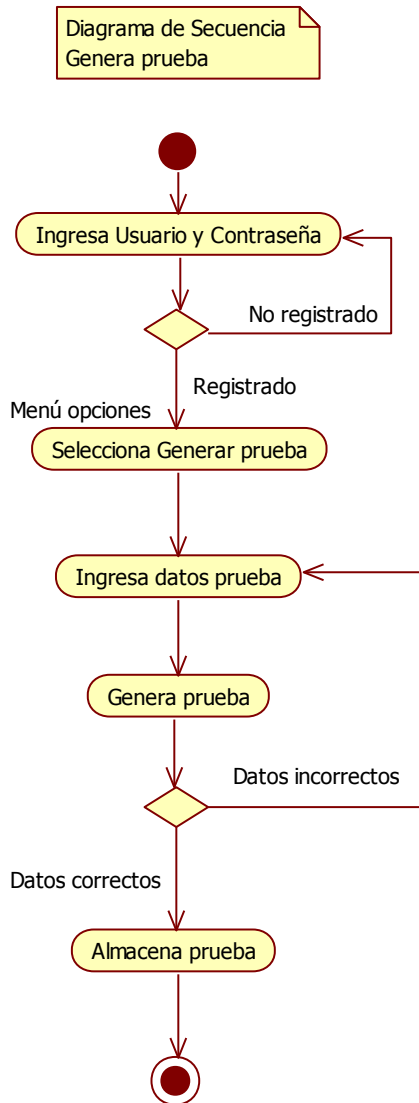


Figura 2.27 Diagrama de actividades
Genera Prueba

2.9.14 Diagrama de actividades Asigna Prueba



Figura 2.28 Diagrama de actividades
Asigna Prueba

2.9.15 Diagrama de actividades Reporte de Calificaciones

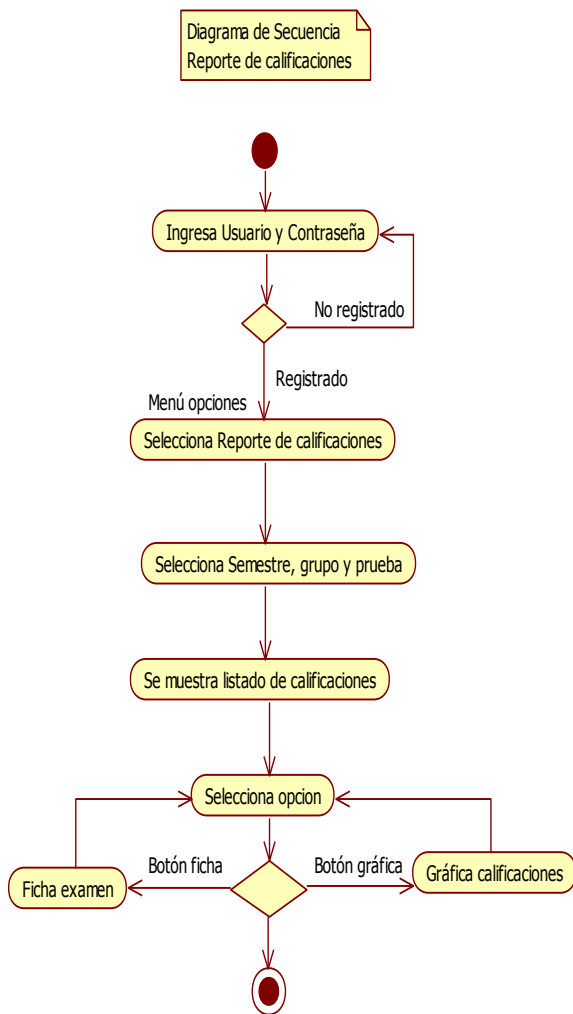


Figura 2.29 Diagrama de actividades Reporte de Calificaciones

2.9.16 Diagrama de actividades Contesta Prueba

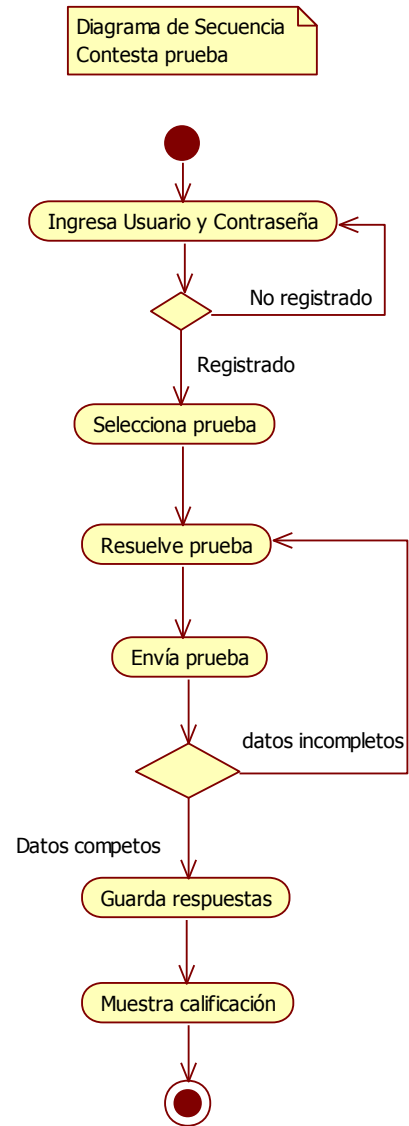


Figura 2.30 Diagrama de actividades Reporte de Contesta Prueba

CAPÍTULO 3. DISEÑO DEL SISTEMA

A continuación definiremos los detalles que establecen la forma en que el sistema cumplirá con los requerimientos identificados durante el análisis.

3.1 Arquitectura del Sistema

Se implementará un servidor Web, el cual almacenará la base de datos y los archivos del sistema. Tras la recepción de una solicitud, el servidor la procesa y envía la respuesta al cliente.

Los clientes hacen su solicitud a través de la web y esperan la respuesta del servidor. Los clientes pueden ser: docente administrador, docentes o alumnos.

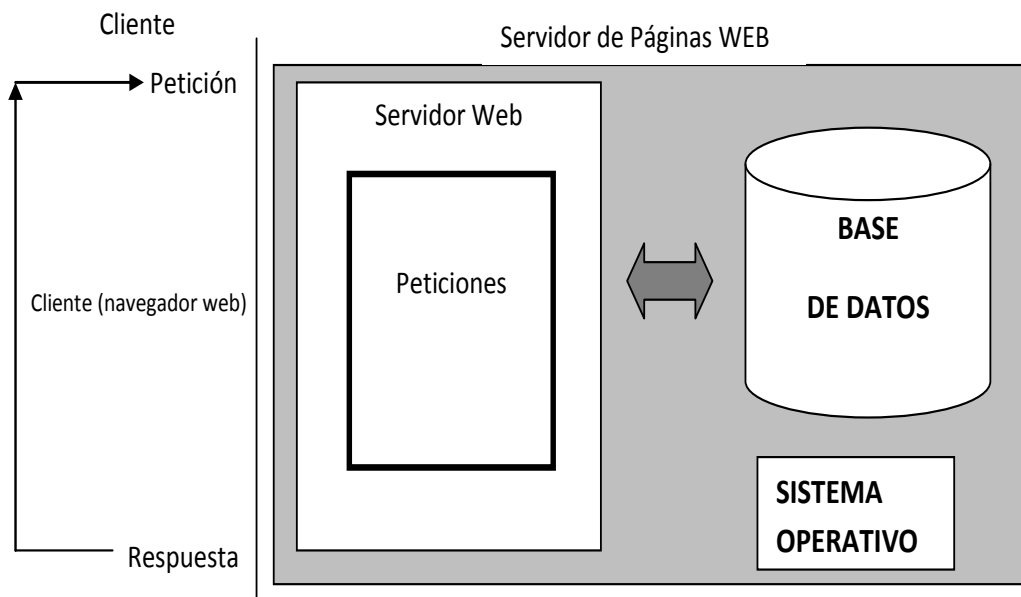


Figura 3.1 Esquema General del Sistema

3.2 Diagramas de navegación del Sistema

En la figura 3.2 se muestra el diagrama que explica la navegación entre los contenidos del Sistema.

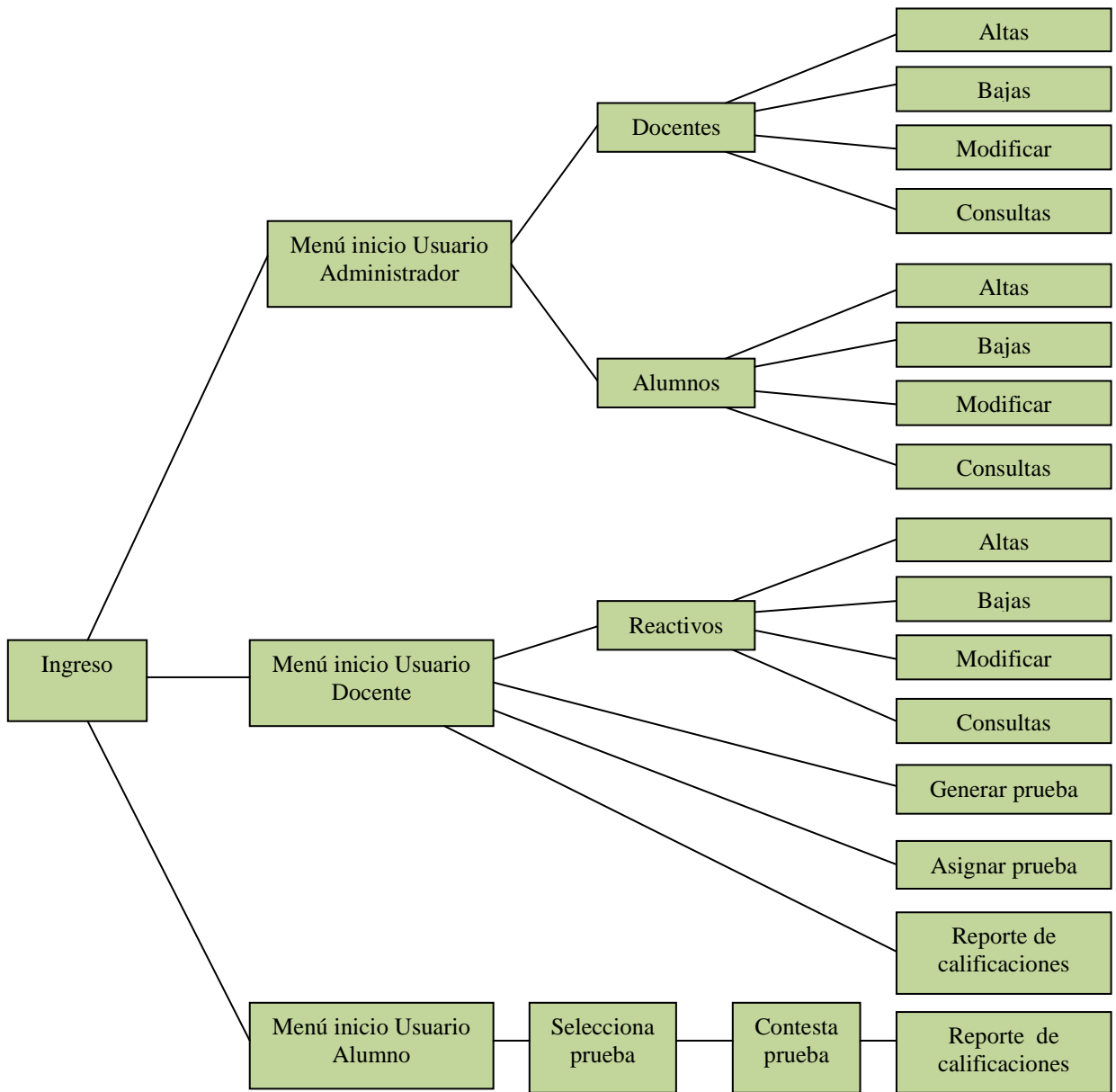


Figura 3.2 Diagrama de Navegación del Sistema

3.3. Diseño de la Base de Datos

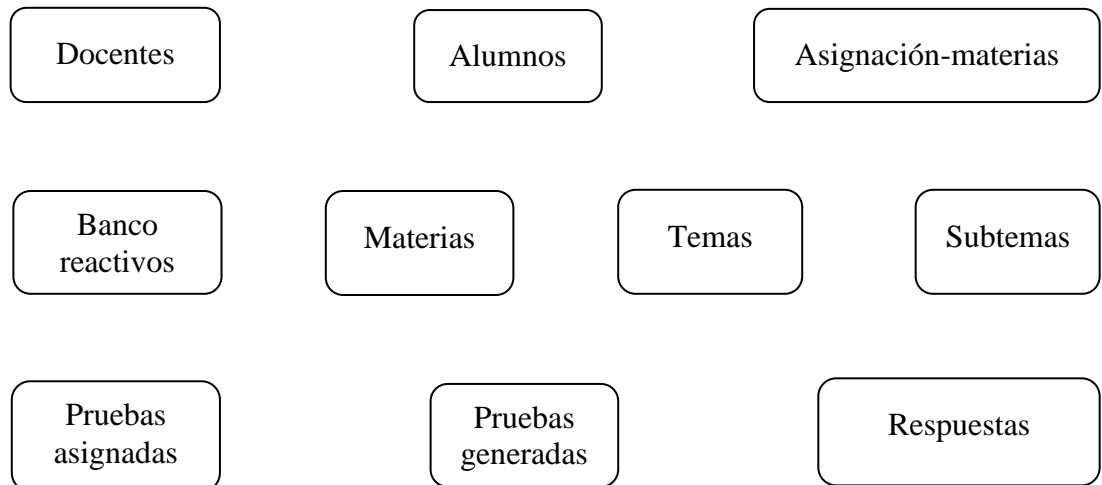
Dado la complejidad de la información al diseñar una base de datos utilizaremos una práctica estándar, el dividir el diseño de base de datos en las siguientes fases:

- Diseño Conceptual
- Diseño Lógico
- Diseño físico

3.3.1 Diseño Conceptual de la Base de Datos

Esta fase se alimenta del análisis de Requerimientos y produce un diseño que trata de reflejar como son los datos.

El modelo conceptual del sistema se compone de las siguientes entidades:



3.3.2 Diagrama Entidad – Relación

Un diagrama o modelo entidad-relación es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

A continuación se muestra el diagrama entidad relación de la base de datos.

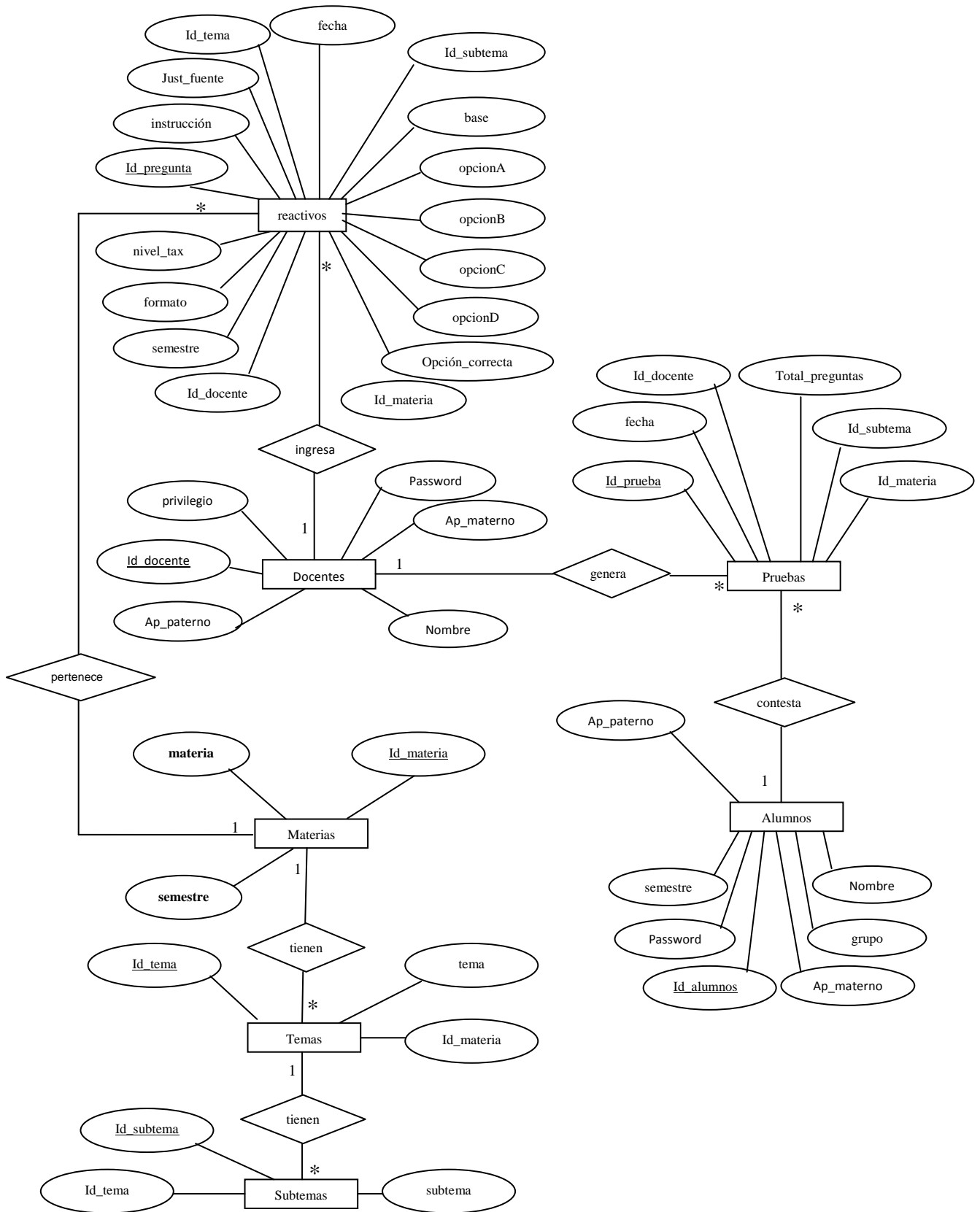


Figura. 3.3 Primera parte Diagrama Entidad - Relación

3.3.3 Diseño Lógico de la Base de Datos

En esta fase se produce un diseño que se acerca más a la implementación en un Sistema Manejador de Base de Datos. Es decir, se transforma el modelo Entidad – Relación en tablas.

3.3.4 Diccionario de Datos

Muestra las características de las entidades y atributos que definen la estructura de la base de datos. A continuación se muestra el diccionario de datos [4]:

DOCENTES

NOMBRE	DESCRIPCIÓN	FORMATO
Id_docente	Clave del docente	Letras y/o números
ap_paterno	Apellido paterno del docente	Caracteres alfanuméricos
ap_materno	Apellido materno del docente	Caracteres alfanuméricos
nombre	Nombre del docente	Caracteres alfanuméricos
Password	Contraseña del docente	Caracteres alfanuméricos
privilegio	nivel de restricción al sistema	1 o 2

ALUMNOS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_alumno	Clave del docente	Letras y/o números
ap_paterno	Apellido paterno del docente	Caracteres alfanuméricos
ap_materno	Apellido materno del docente	Caracteres alfanuméricos
nombre	Nombre del docente	Caracteres alfanuméricos
semestre	Semestre en el que se encuentra inscrito el alumno	1 a.8
grupo	Contraseña del docente	A o B
password	nivel de restricción al sistema	Caracteres alfanuméricos

MATERIAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_materia	Clave del docente	Letras y/o números
materia	Apellido paterno del docente	Caracteres alfanuméricos
semestre	Apellido materno del docente	1 a.8

ASIGNACIÓN_MATERIAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_docente	Clave del docente	Letras y/o números
Id_materia	Identificador de la materia	Letras y/o números
grupo	Grupo en el que el docente imparte la asignatura	A o B

BANCO_REACTIVOS

NOMBRE	DESCRIPCIÓN	FORMATO
id_preg	Identificador de la pregunta	Letras y/o números
instrucción	Instrucciones para contestar la pregunta	Caracteres alfanuméricos
Base	Texto de la Pregunta	Caracteres alfanuméricos
opcionA	respuesta posible que contesta correctamente la pregunta	Caracteres alfanuméricos
opcionB	respuesta posible que contesta correctamente la pregunta	Caracteres alfanuméricos
opcionC	respuesta posible que contesta correctamente la pregunta	Caracteres alfanuméricos
opcionD	respuesta posible que contesta correctamente la pregunta	Caracteres alfanuméricos
opción_correcta	Opción que contesta correctamente la pregunta	A o B C o D
Formato		Caracteres alfanuméricos
nivel_tax		Caracteres alfanuméricos
just_fuente		Caracteres alfanuméricos
id_tema	Identificador del tema	Caracteres alfanuméricos
id_subtema	Identificador del subtema	Caracteres alfanuméricos

id_materia	Identificador de la materia	Caracteres alfanuméricos
id_docente	Identificador del docente que elaboró el reactivo	Letras y/o números
Fecha	Fecha en que se elaboró el reactivo	Año/mes/día

PRUEBAS_GENERADAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_examen	Identificador de la prueba	Caracteres alfanuméricos
Id_materia	Identificador de la materia	Caracteres alfanuméricos
Id_subtema	Identificador del subtema	Caracteres alfanuméricos
total_preg	Cantidad de preguntas que se mostrarán del tema	Número entero
Id_docente	Identificador del docente que generó la prueba	Letras y/o números
Fecha	Fecha en que se generó la prueba	Año/mes/día

PRUEBAS_ASIGNADAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_alumno	Identificador del alumno	Letras y/o números
Id_examen	Identificador de la prueba	Caracteres alfanuméricos
fecha	Fecha en que se generó la prueba	Año/mes/día

RESPUESTAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_examen	Identificador de la prueba	Caracteres alfanuméricos
Id_alumno	Identificador del alumno	Letras y/o números
respuestas	Respuestas que eligió el alumno	Cadena de caracteres A,B,C o D
resp_correctas	Respuestas que contestan correctamente cada pregunta	Cadena de caracteres A,B,C o D
preguntas	contiene todos los identificadores de las preguntas mostradas	Caracteres alfanuméricos
fecha	Fecha en que se generó la prueba	Date

TEMAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_tema	Identificador del tema	Caracteres alfanuméricos
tema	Nombre del tema	Caracteres alfanuméricos
Id_materia	Identificador de la materia	Caracteres alfanuméricos

SUBTEMAS

NOMBRE	DESCRIPCIÓN	FORMATO
Id_subtema	Identificador del subtema	Caracteres alfanuméricos
subtema	Nombre del subtema	Caracteres alfanuméricos
Id_tema	Identificador del tema	Caracteres alfanuméricos

3.3.5 Diseño lógico.

Se muestra el diseño lógico del sistema, resultado de transformar el modelo conceptual del sistema.

DOCENTES				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_docente	Clave del docente	Varchar(14)	No	Si
ap_paterno	Apellido paterno del docente	Text	No	
ap_materno	Apellido materno del docente	Text	No	
nombre	Nombre del docente	Text	No	
Password	Contraseña del docente	Varchar(20)	No	
privilegio	nivel de restricción al sistema	Varchar(1)	No	

ALUMNOS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_alumno	Clave del docente	Varchar(14)	No	Si
ap_paterno	Apellido paterno del docente	Text	No	
ap_materno	Apellido materno del docente	Text	No	
nombre	Nombre del docente	Text	No	
semestre	Semestre en el que se encuentra inscrito el alumno	int(1)	No	
grupo	Contraseña del docente	Varchar(20)	No	
password	nivel de restricción al sistema	Varchar(1)	No	

MATERIAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_materia	Clave del docente	Varchar(10)	No	Si
materia	Apellido paterno del docente	Text	No	
semestre	Apellido materno del docente	Int(1)	No	

ASIGNACION_MATERIAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_docente	Clave del docente	Varchar(14)	No	Si
Id_materia	Identificador de la materia	Varchar(10)	No	
grupo	Grupo en el que el docente imparte la asignatura	Char(1)	No	

BANCO_REACTIVOS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
id_preg	Identificador de la pregunta	Varchar(14)	No	Si
instrucción	Instrucciones para contestar la pregunta	Text	Si	
base	Texto de la Pregunta	Text	No	
opcionA	respuesta posible que contesta correctamente la pregunta	Text	No	
opcionB	respuesta posible que contesta correctamente la pregunta	Text	No	
opcionC	respuesta posible que contesta correctamente la pregunta	Text	No	
opcionD	respuesta posible que contesta correctamente la pregunta	Text	No	
opción_correcta	Opción que contesta correctamente la pregunta	Char(1)	No	
formato		Text	No	
nivel_tax		Text	No	
just_fuente		Text	No	
id_tema	Identificador del tema	Varchar(20)	No	Si
id_subtema	Identificador del subtema	Varchar(20)	No	Si
id_materia	Identificador de la materia	Varchar(14)	No	Si
id_docente	Identificador del docente que elaboró el reactivo	Varchar(14)	No	Si
fecha	Fecha en que se elaboró el reactivo	Date	No	

PRUEBAS_GENERADAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_examen	Identificador de la prueba	Varchar(50)	No	Si
Id_materia	Identificador de la materia	Varchar(14)	No	Si
Id_subtema	Identificador del subtema	Varchar(20)	No	Si
total_preg	Cantidad de preguntas que se mostrarán del tema	Int(11)	No	
Id_docente	Identificador del docente que generó la prueba	Varchar(14)	No	Si
Fecha	Fecha en que se generó la prueba	date	No	

PRUEBAS_ASIGNADAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_alumno	Identificador del alumno	Varchar(14)	No	Si
Id_examen	Identificador de la prueba	Varchar(50)	No	Si
Fecha	Fecha en que se generó la prueba	Date	No	

RESPUESTAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_examen	Identificador de la prueba	Varchar(50)	No	Si
Id_alumno	Identificador del alumno	Varchar(14)	No	Si
respuestas	Respuestas que eligió el alumno	Text	No	
resp_correctas	Respuestas que contestan correctamente cada pregunta	Text	No	
preguntas	contiene todos los	Text	No	

fecha	identificadores de las preguntas mostradas Fecha en que se generó la prueba	Date	No	
-------	--	------	----	--

TEMAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_tema	Identificador del tema	Varchar(20)	No	Si
tema	Nombre del tema	Text	No	
Id_materia	Identificador de la materia	Varchar(10)	No	Si

SUBTEMAS				
CAMPO	DESCRIPCIÓN	TIPO	NULO	LLAVE
Id_subtema	Identificador del subtema	Varchar(20)	No	Si
subtema	Nombre del subtema	Text	No	
Id_tema	Identificador del tema	Varchar(10)	No	Si

3.3.6 Normalización de la Base de Datos

Existen básicamente tres niveles de normalización: Primera Forma Normal (1NF), Segunda Forma Normal (2NF) y Tercera Forma Normal (3NF). Cada una de estas formas tiene sus propias reglas.

Las formas normales garantizan que varios de los problemas de redundancia y de anomalías en una base de datos no ocurran.

Normalización de la base de datos Docentes

id_docente (llave primaria)	ap_paterno	ap_materno	nombre	password	Privilegio
agarcia	García	Aguilar	Adolfo	agarcia	2
ahernandez	Hernández	Flores	Abel	ahernandez	2
tjimenez	Jiménez	León	Teresa	tjimenez	2

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_docente** ya que cualquier valor único de **id_docente** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos Alumnos

id_alumno (llave primaria)	ap_paterno	ap_materno	Nombre	semestre	grupo	password
03201205	Liliana	Merino	García	2	A	1
2111156910	Aragón	Vivas	Antonio	3	B	1
212008002	Alatorre	Cuenca	Concepción	2	A	1

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_alumno** ya que cualquier valor único de **id_alumno** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos Materias

id_materia (llave primaria)	materia	semestre
ART2	Educación Artística II	5
ART3	Educación Artística III	6
ESP1	Español y su Enseñanza I	2

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.

- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_materia** ya que cualquier valor único de **id_materia** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos **Asignación_materias**

Id_asignacion (llave primaria)	id_docente	Id_materia	grupo
1	ssaavedra	ART2	B
2	agarcia	HIST2	A
3	tjimenez	MAT1	B

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_asignacion** ya que cualquier valor único de **id_asignacion** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos **banco_reactivos**

id_preg (llave primaria)	instrucción	base	opcionA	opcionB	opcionC	opcionD	opción_ correcta	Formato
MAT11		En cuál de los siguientes casos el resultado de la...	El pizarrón es tres veces más grande que la mesa	Mi mochila es más pesada que la de Juan	Juan es doble de gordo que Pedro	Pedro mide el doble de largo que su hermano	B	Cuestionamiento directo
MAT12		En la realización de un experimento , la maestra de...	Directa	Indirecta	Formal	Informal	A	Cuestionamiento directo
MAT15		Cuál de las siguientes respuestas no corresponde c...	Plantear una meta comprensible para quien la va a ...	Permitir aproximaciones a la solución a partir de ...	Plantear un reto, una dificultad.	ser muy fácil, no debe presentar ninguna dificultad...	D	Cuestionamiento directo

nivel_tax	just_fuente	id_tema	id_subtema	id_materia	id_docente	fecha
Conocimiento o memoria	B es la opción correcta porque es la única opción ...	MAT1T1	MAT1T1ST5	MAT1	tjimenez	2012-10-23
Conocimiento o memoria	La opción correcta es A por que para realizar la c...	MAT1T1	MAT1T1ST5	MAT1	tjimenez	2012-10-23
Conocimiento o memoria	La opción D es la respuesta correcta pues los probl...	MAT1T1	MAT1T1ST2	MAT1	tjimenez	2012-10-23

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_preg** ya que cualquier valor único de **id_preg** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos pruebas_generadas

Id_prueba (llave primaria)	Id_materia	Id_subtema	total_preg	Id_docente	Fecha
Primer momento_24/10/2012	MAT1	MAT1T1ST2	20	tjimenez	2012-10-24
Semanal 1_25/10/2012	MAT1	MAT1T1ST2	10	tjimenez	2012-10-25
Segundo Momento_05/11/2012	MAT1	MAT1T1ST5	20	tjimenez	2012-11-05

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_prueba** ya que cualquier valor único de **id_prueba** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos Pruebas Asignadas

Id_prueba_asig (llave primaria)	Id_alumno	Id_examen	Fecha
1	20110001	Primer momento_24/10/2012	2012-10-24
2	03201204	Semanal 1_25/10/2012	2012-10-25
3	21300009	Semanal 1_25/10/2012	2012-10-25

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- Podemos observar que la tabla se encuentra en 1FN
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_prueba_asig** ya que cualquier valor único de **id_prueba_asig** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos Respuestas

Id_respuestas (llave primaria)	Id_examen	Id_alumno	respuestas	resp_correctas	preguntas	Fecha
1	Primer momento_24/10/2012	21201005	A,A,D,A,D, ...	D,A,C,A,B,...	MAT15,MAT14, MAT13,MAT12, MAT11,...	2012-11-13
2	Primer momento_24/10/2012	21201003	D,D,D,D,B, ...	D,D,A,C,B,	MAT16,MAT15, MAT12,MAT13, MAT11,...	2012-11-13
3	Primer momento_24/10/2012	21201004	A,C,B,C,C, ...	A,D,B,A,C,...	MAT14,MAT15, MAT11,MAT12, MAT13,...	2012-11-13

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_respuestas** ya que cualquier valor único de **id_respuestas** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos Temas

Id_tema (llave primaria)	tema	Id_materia
MAT1T1	Aprender matemáticas al resolver problemas	MAT1
MAT1T2	Los números naturales y el sistema decimal de nume...	MAT1
MAT1T3	Las cuatro operaciones básicas con números natural...	MAT1

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_tema** ya que cualquier valor único de **id_tema** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

Normalización de la base de datos Subtemas

Id_subtema (llave primaria)	subtema	id_tema
MAT1T1ST1	Nuestra experiencia personal con las matemáticas	MAT1T1
MAT1T1ST2	El papel de los problemas en el aprendizaje de las...	MAT1T1
MAT1T1ST3	El papel del juego en el aprendizaje de las matemá...	MAT1T1

- Podemos observar que la tabla se encuentra en 1FN. Cada valor esta dentro de su rango de valores y existe un solo valor para cada atributo de la relación.
- La tabla también se encuentra en 2FN. Todos los campos no clave son dependientes de la llave primaria **id_subtema** ya que cualquier valor único de **id_alumno** determina un sólo valor para cada columna.
- Ningún atributo no clave es funcionalmente dependiente de algún otro atributo no clave, por tanto la tabla se encuentra en 3FN.

A continuación, en la figura 3.5, se muestra el modelo Relacional. Los datos están estructurados a nivel lógico como tablas formadas por filas y columnas, aunque a nivel físico pueden tener una estructura distinta.

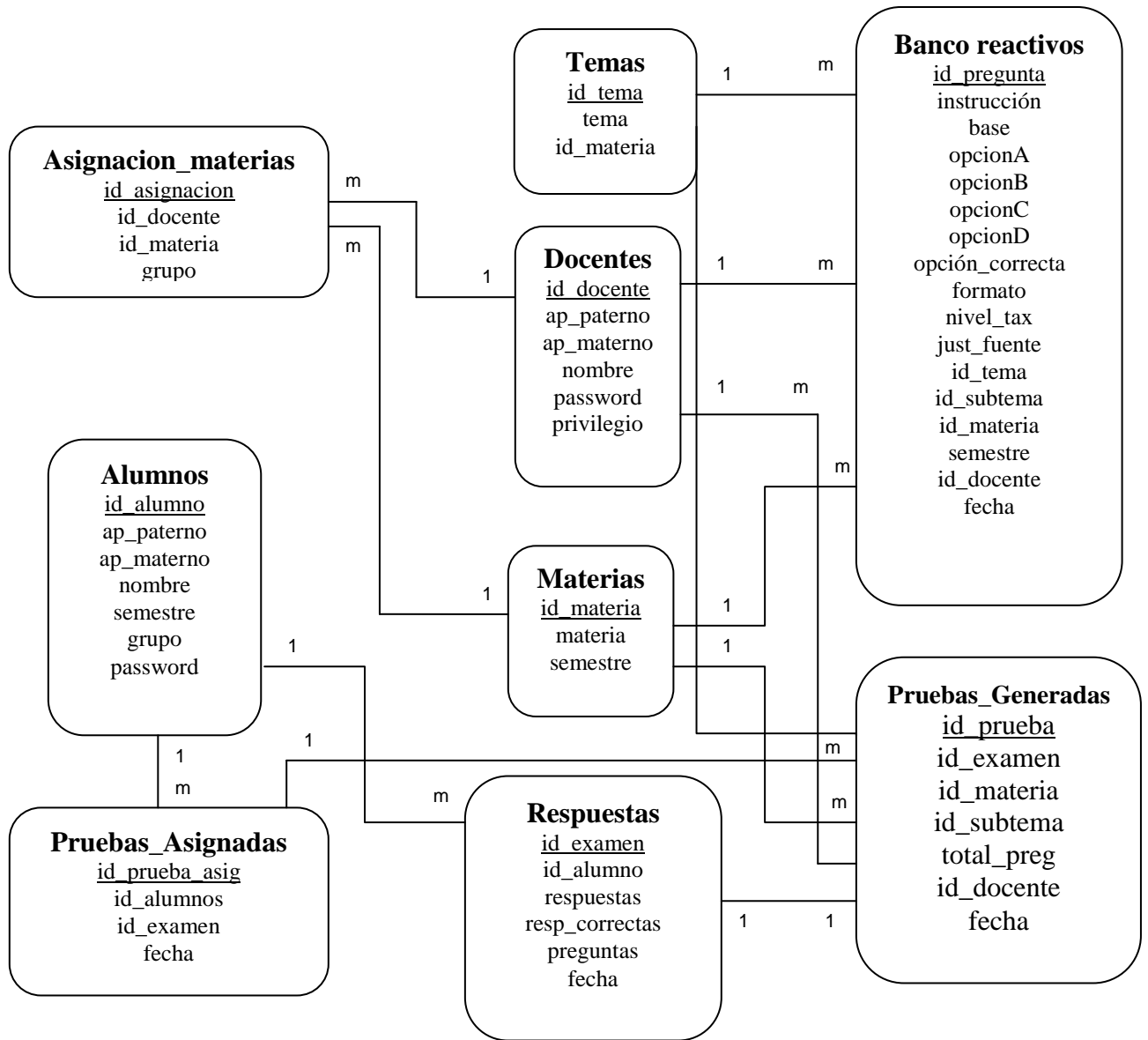


Figura. 3.5 Diagrama Entidad - Relación

3.3.7 Diseño Físico de la Base de Datos

El Diseño físico está construido sobre la base del modelo lógico y describe como los datos son almacenados en el Sistema manejador de bases de datos MySQL.

```
CREATE TABLE `docentes` (  
  `id_docente` varchar(14) NOT NULL,  
  `ap_paterno` text NOT NULL,  
  `ap_materno` text NOT NULL,  
  `nombre` text NOT NULL,  
  `password` varchar(20) NOT NULL,  
  `privilegio` int(1) NOT NULL,  
  PRIMARY KEY (`id_docente`)  
)
```

```
CREATE TABLE `alumnos` (  
  `id_alumno` varchar(14) NOT NULL,  
  `ap_paterno` text NOT NULL,  
  `ap_materno` text NOT NULL,  
  `nombre` text NOT NULL,  
  `semestre` int(1) NOT NULL,  
  `grupo` varchar(2) NOT NULL,  
  `password` varchar(20) NOT NULL,  
  PRIMARY KEY (`id_alumnos`)  
)
```

```
CREATE TABLE `banco_reactivos` (  
  `id_pregunta` varchar(14) NOT NULL,  
  `instruccion` text NOT NULL,  
  `base` text NOT NULL,  
  `opcionA` text NOT NULL,  
  `opcionB` text NOT NULL,  
  `opcionC` text NOT NULL,  
  `opcionD` text NOT NULL,  
  `opcion_correcta` char(1) NOT NULL,  
  `formato` text NOT NULL,  
  `nivel_tax` text NOT NULL,  
  `just_fuente` text NOT NULL,  
  `id_tema` varchar(20) NOT NULL,  
  `id_subtema` varchar(20) NOT NULL,  
  `id_materia` varchar(14) NOT NULL,  
  `semestre` int(1) NOT NULL,  
  `id_docente` varchar(14) NOT NULL,  
  `fecha` date NOT NULL,  
  PRIMARY KEY (`id_pregunta`)  
)
```

```
CREATE TABLE `materias` (  
  `id_materia` varchar(10) NOT NULL,  
  `materia` text NOT NULL,  
  `semestre` int(2) NOT NULL,  
  PRIMARY KEY (`id_materia`)  
)
```

```
CREATE TABLE `asignacion_materias` (  
  `id_asignacion` int(2) NOT NULL,  
  `id_docente` varchar(14) NOT NULL,  
  `id_materia` varchar(10) NOT NULL,  
  `grupo` char(1) NOT NULL,  
  PRIMARY KEY (`id_asignacion`)  
)
```

```
CREATE TABLE `pruebas_generadas` (  
  `id_prueba` int(7) NOT NULL,  
  `id_examen` varchar(50) NOT NULL,  
  `id_materia` varchar(10) NOT NULL,  
  `id_subtema` varchar(20) NOT NULL,  
  `total_preg` int(2) NOT NULL,  
  `id_docente` varchar(14) NOT NULL,  
  `fecha` date NOT NULL,  
  PRIMARY KEY (`id_prueba`)  
)
```

```
CREATE TABLE `pruebas_asignadas` (  
  `id_prueba_asig` int(7) NOT NULL,  
  `id_alumnos` varchar(14) NOT NULL,  
  `id_examen` varchar(50) NOT NULL,  
  `fecha` date NOT NULL,  
  PRIMARY KEY (`id_prueba_asig`)  
)
```

```
CREATE TABLE `respuestas` (  
  `id_examen` varchar(50) NOT NULL,  
  `id_alumno` varchar(14) NOT NULL,  
  `respuestas` text NOT NULL,  
  `resp_correctas` text NOT NULL,  
  `preguntas` text NOT NULL,  
  `fecha` date NOT NULL,  
  PRIMARY KEY (`id_examen`,`id_alumno`)  
)
```

```
CREATE TABLE `subtemas` (  
  `id_subtema` varchar(20) NOT NULL,  
  `subtema` text NOT NULL,  
  `id_tema` varchar(20) NOT NULL,
```

```
PRIMARY KEY (`id_subtema`)  
)
```

```
CREATE TABLE `temas` (  
  `id_tema` varchar(20) NOT NULL,  
  `tema` text NOT NULL,  
  `id_materia` varchar(10) NOT NULL,  
  PRIMARY KEY (`id_tema`)  
)
```

CAPITULO 4. DESARROLLO Y PRUEBAS DEL SISTEMA

4.1 Ambiente de Desarrollo

Para el desarrollo de la aplicación se tomó la decisión de que las pruebas aplicadas fueran del tipo objetivas conocidas también como de opción múltiple, ya que este tipo de pruebas permiten hacer inferencias, estudios estadísticos, graficas y análisis para comparar los resultados entre los estudiantes de un grupo o de diferentes grupos, así como comparar los resultados de cada estudiante con los objetivos curriculares alcanzados, se pueden aplicar a gran escala o a grupos numerosos,; su forma de calificar es relativamente sencilla utilizando Software Informático, además con estas, se reducen notablemente los efectos subjetivos de la percepción personal del evaluador en los resultados.

Fue necesaria la participación de algunos docentes para la elaboración de reactivos con el nivel cognoscitivo propuesta por Benjamín S. Bloom

4.2 Herramientas de Desarrollo.

Las herramientas que se requirieron para el desarrollo de la aplicación son las siguientes:

- ADOBE DREAMWEAVER 8
- MYSQL 5.0.18-log
- REACTOR con phpMyAdmin 2.70-pl2
- NAVEGADOR DE PAGINAS WEB Firefox v.3.0 o superior
- servidor apache versión 2.2.8 y phpMyAdmin 2.70-pl2

Se requirió también las librerías phplot.php y rgb.inc.php, que permiten al sistema generar gráficas.

Primeramente se instaló el servidor apache con PHP y MySQL que viene incluido en el software REACTOR ya que toda la programación y pruebas se realizaron de manera local.

Para facilitar la programación, diseño y edición de las páginas web, se utilizó el software adobe dreamweaver 8, ya que ofrece una vista preliminar de la página (no requiere de un navegador para mostrar las paginas) y además permite dar formato a la página de una forma homogénea a través del manejo de las hojas de estilos.

Igualmente se requirió de un navegador de páginas web como mozilla versión 3.0 o superior para poder ir visualizando la aplicación y para las pruebas del sistema durante el desarrollo del mismo.

4.3 Conexión y acceso a la Base de Datos del Sistema.

Para el acceso a la base de datos del sistema debe estar ejecutándose en el servidor la aplicación Reactor. La conexión a la base de datos de **MySQL** se hace desde PHP a través de las funciones PHP *mysql_connect* y *mysql_select_db*. Para realizar las modificaciones y consultas a la base de datos se utiliza el lenguaje estándar de consultas estructurado (SQL), el cual facilita el acceso a la base de datos de MySQL.

.

4.4 Requerimientos de Hardware.

El hardware mínimo que se requirió para el desarrollo de la aplicación es:

- Equipo con procesador Pentium IV
- 512 megabytes en RAM.
- unidad lectora de CD ó DVD.

Para la fase de desarrollo del proyecto, se utilizó el lenguaje de programación PHP, HTML, JavaScript, Ajax y herramientas para el diseño de las pantallas.

A continuación se muestran las principales pantallas que componen sistema.

4.5 Pantallas del Sistema

El sistema está conformado por varias pantallas. Existe una pantalla principal de nombre Menú la cual contiene todas las opciones que el usuario Administrador y usuario Docente pueden acceder.

A continuación se muestran las principales pantallas del sistema.

4.5.1 Pantalla de Ingreso al Sistema

Es a través de esta pantalla que los usuarios pueden ingresar al sistema. Para ello deben ingresar su nombre de usuario y una contraseña.

En caso de que el usuario ingrese incorrectamente su nombre de usuario o su contraseña se emite un mensaje de error y se le solicita nuevamente la información.

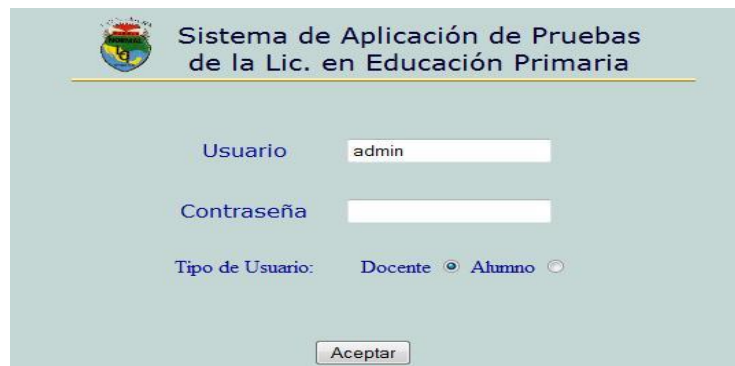


Figura 4.1 Login del Sistema

4.5.2 Pantalla Menú de Opciones

En esta pantalla se muestran todas las opciones del Sistema



Figura 4.2 Menú Opciones

4.5.3 Pantalla Ingresar Docentes

En esta pantalla se capturan los datos de los docentes

Ingresar Docentes

Apellido Paterno: Ramirez

Apellido Materno: Alcaide

Nombre: Juan

Usuario: jramirez

Contraseña: ●●●●●●

Privilegios: Docente Administrador

GUARDAR REGRESAR

Figura 4.3 Ingresar Docentes

4.5.4 Pantalla Ingresar Alumnos

En esta pantalla se capturan los datos de los alumnos

Ingresar Alumnos

Apellido Paterno: Rivera

Apellido Materno: Carrillo

Nombre: Christian Jesús

Matrícula: 03201212

Semestre: Tercer Semestre

Grupo: "A"

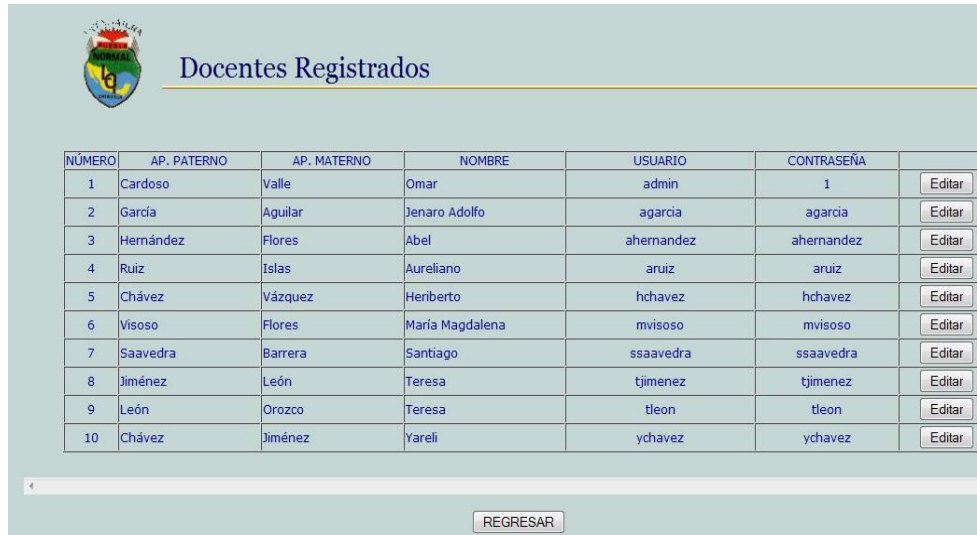
Contraseña: |

GUARDAR REGRESAR

Figura 4.4 Ingresar Alumnos

4.5.5 Pantalla Docentes registrados

Esta pantalla muestran los docentes registrados. A través de esta pantalla podemos editar los datos de un docente.



NÚMERO	AP. PATERNO	AP. MATERNO	NOMBRE	USUARIO	CONTRASEÑA	Editar
1	Cardoso	Valle	Omar	admin	1	Editar
2	García	Aguiar	Jenaro Adolfo	agarcia	agarcia	Editar
3	Hernández	Flores	Abel	ahernandez	ahernandez	Editar
4	Ruiz	Islas	Aureliano	aruiz	aruiz	Editar
5	Chávez	Vázquez	Heriberto	hchavez	hchavez	Editar
6	Visoso	Flores	María Magdalena	mvisoso	mvisoso	Editar
7	Saaavedra	Barrera	Santiago	ssaavedra	ssaavedra	Editar
8	Jiménez	León	Teresa	tjimenez	tjimenez	Editar
9	León	Orozco	Teresa	tleon	tleon	Editar
10	Chávez	Jiménez	Yareli	ychavez	ychavez	Editar

Figura 4.5 Docentes registrados

4.5.6 Pantalla Busca alumnos

Esta pantalla nos permite buscar a los alumnos registrados. A través de esta pantalla podemos editar los datos de un alumno.



NÚMERO	MATRÍCULA	AP. PATERNO	AP. MATERNO	NOMBRE	CONTRASEÑA	Editar
1	21201001	Acevedo	Carrilo	Nicte Libertad	acevedo	Editar
2	21201002	Aguiar	Zacatenco	Teresa	aguiar	Editar
3	21201003	Alonso	Solis	Amy Ilse	alonso	Editar
4	21201004	Alvarado	Tlaltenchi	Teresa	alvarado	Editar
5	21201005	Aragón	Estudillo	José Juan	aragon	Editar
6	21201006	Balbuena	Barreto	Elidey	balbuena	Editar

Figura 4.6 Buscar Alumnos

4.5.7 Pantalla Acta de Nacimiento del Reactivo

Es en esta pantalla donde se ingresan las preguntas en el banco de reactivos.

Acta de Nacimiento del Reactivo

Instrucción
Selecciona la opción que conteste correctamente la pregunta

Base
¿Cuál de las siguientes afirmaciones NO ES CORRECTA?

RESPUESTAS

Opción A
La búsqueda de la solución a un problema nuevo empieza muchas veces por tanteos, ensayos, errores y correcciones

Opción B
Antes de plantearse el problema a una persona se le debe enseñar la "fórmula" que lo resuelve de manera sistemática.

Opción C
Es al resolver problemas cuando los alumnos pueden construir sus conocimientos matemáticos de manera que éstos tengan significación para ellos.

Opción D
Una serie puede utilizarse para comparar colecciones. Basta con conocer bien el orden de los elementos de la serie.

Opción Correcta: "B"

Formato
Cuestionamiento directo

Nivel Taxonómico
Conocimiento o memoria

Justificación y Fuente
La opción "B" es la respuesta correcta. El nuevo enfoque refiere que las formulas es lo último que se le debe enseñar a un niño por tanto es una afirmación incorrecta.

IDENTIFICADORES DEL REACTIVO

Semestre:

Materia:

Tema:

Subtema:

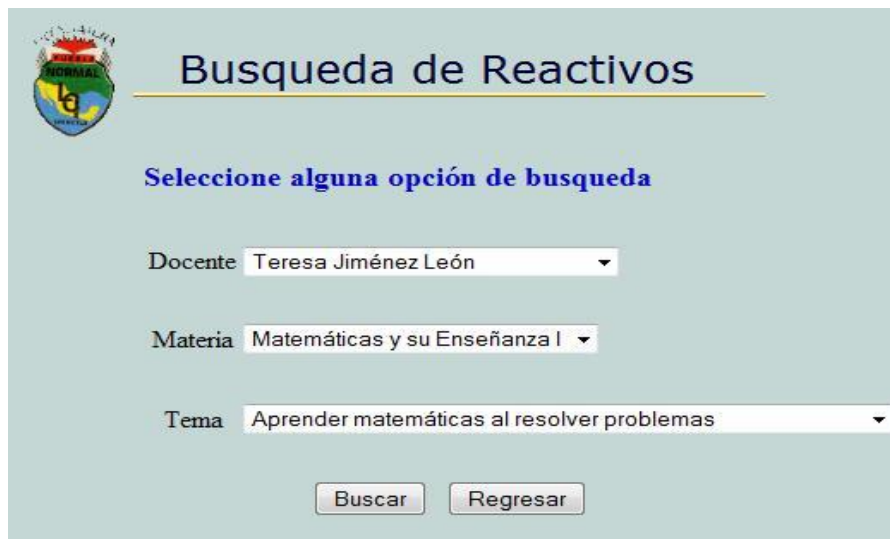
Fecha:

Elaborador:

Figura 4.7/ Buscar Alumnos

4.5.8 Pantalla Busca Reactivos

Esta pantalla nos permite buscar los reactivos almacenados en el banco de reactivos



Busqueda de Reactivos

Seleccione alguna opción de búsqueda

Docente: Teresa Jiménez León

Materia: Matemáticas y su Enseñanza I

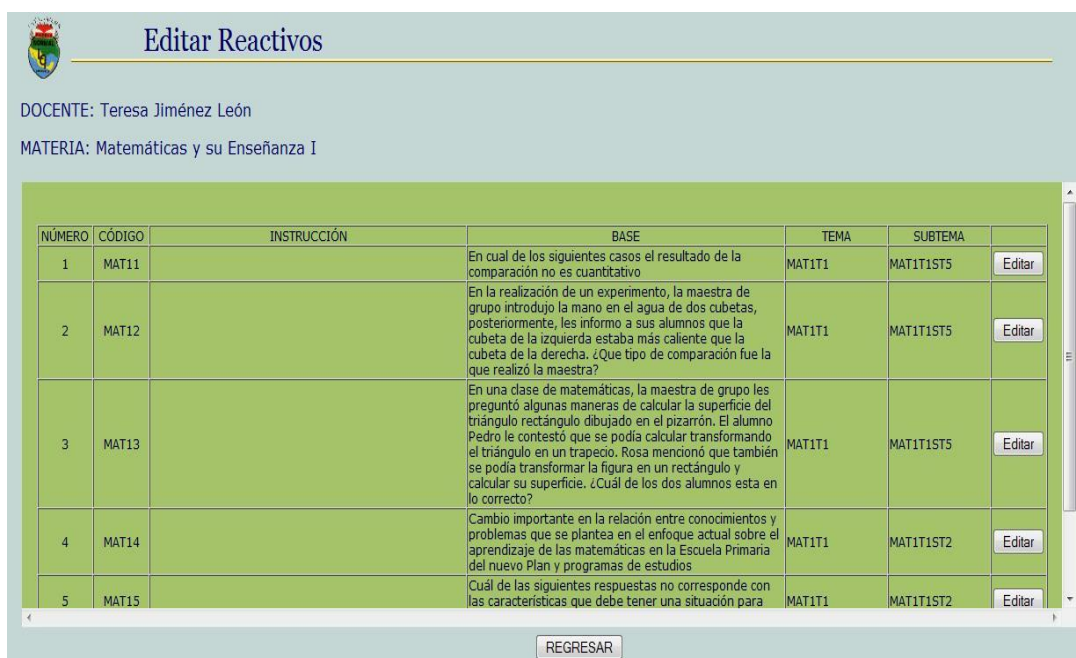
Tema: Aprender matemáticas al resolver problemas

Buscar Regresar

Figura 4.8 Buscar Reactivos

4.5.9 Pantalla Edita Reactivos

Esta pantalla muestra todos los reactivos encontrados. Nos permite editar los reactivos.



Editar Reactivos

DOCENTE: Teresa Jiménez León

MATERIA: Matemáticas y su Enseñanza I

NÚMERO	CÓDIGO	INSTRUCCIÓN	BASE	TEMA	SUBTEMA	
1	MAT11		En cual de los siguientes casos el resultado de la comparación no es cuantitativo	MAT1T1	MAT1T1ST5	Editar
2	MAT12		En la realización de un experimento, la maestra de grupo introdujo la mano en el agua de dos cubetas, posteriormente, les informo a sus alumnos que la cubeta de la izquierda estaba más caliente que la cubeta de la derecha. ¿Que tipo de comparación fue la que realizó la maestra?	MAT1T1	MAT1T1ST5	Editar
3	MAT13		En una clase de matemáticas, la maestra de grupo les preguntó algunas maneras de calcular la superficie del triángulo rectángulo dibujado en el pizarrón. El alumno Pedro le contestó que se podía calcular transformando el triángulo en un trapecio. Rosa mencionó que también se podía transformar la figura en un rectángulo y calcular su superficie. ¿Cuál de los dos alumnos esta en lo correcto?	MAT1T1	MAT1T1ST5	Editar
4	MAT14		Cambio importante en la relación entre conocimientos y problemas que se plantea en el enfoque actual sobre el aprendizaje de las matemáticas en la Escuela Primaria del nuevo Plan y programas de estudios	MAT1T1	MAT1T1ST2	Editar
5	MAT15		Cuál de las siguientes respuestas no corresponde con las características que debe tener una situación para	MAT1T1	MAT1T1ST2	Editar

REGRESAR

Figura 4.9 Editar Reactivos

4.5.10 Pantalla Generar Prueba

En esta pantalla podemos generar las pruebas que se aplicarán a los alumnos.

Crear Prueba

Nombre de la Prueba:

Selecciona el semestre:

Selecciona Materia:

Selecciona el Tema:

Selecciona el Subtema:

Num. de Preguntas:

Código	Materia	Tema	Sub Tema	Preg.	
MAT1T1ST2	Matemáticas y su Enseñanza I	Aprender matemáticas al resolver problemas	El papel de los problemas en el aprendizaje de las matemáticas	6	<input type="button" value="Eliminar"/>

Figura 4.10 Generar Prueba

4.5.11 Pantalla Asignar Prueba

Es en esta pantalla donde los docentes pueden asignar las pruebas (que generaron) a sus alumnos.

Asignar Prueba

Seleccione un Docente:

Seleccione el semestre:

Seleccione el Grupo:

Seleccione la Materia:

Exámenes creados:

Figura 4.11 Asignar Prueba

4.5.12 Pantalla Consultar Calificaciones

Esta pantalla permite consultar las calificaciones obtenidas por los alumnos de un examen específico.



Docente: Teresa Jiménez León

Materia: Matemáticas y su Enseñanza I

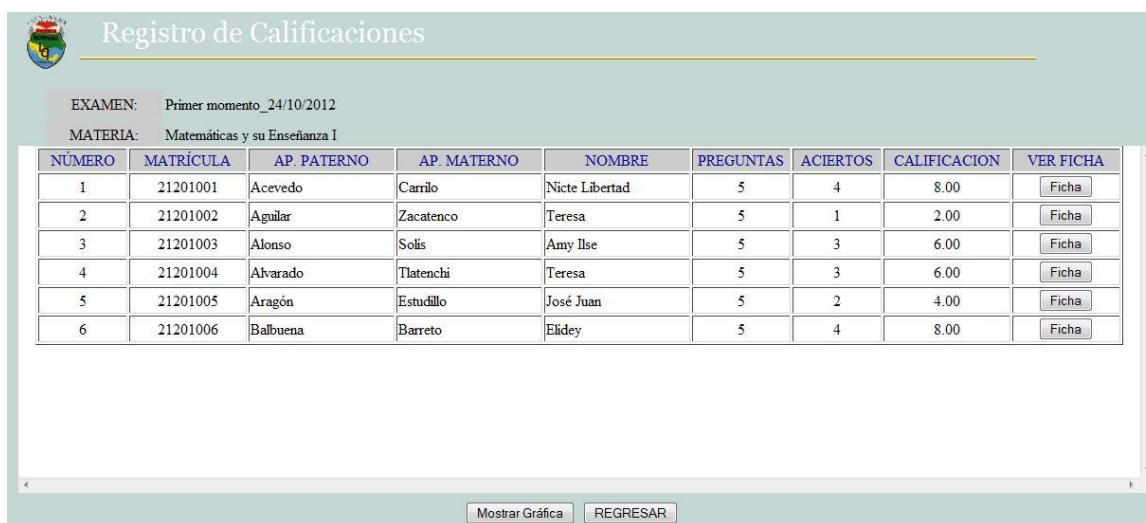
Examen: Primer momento_24/10/2012

ACEPTAR REGRESAR

Figura 4.12 Consultar Calificaciones

4.5.13 Pantalla Registro de Calificaciones

Nos muestra el registro de calificaciones obtenidas por los alumnos que contestaron la prueba.



EXAMEN: Primer momento_24/10/2012

MATERIA: Matemáticas y su Enseñanza I

NÚMERO	MATRÍCULA	AP. PATERNO	AP. MATERNO	NOMBRE	PREGUNTAS	ACIERTOS	CALIFICACION	VER FICHA
1	21201001	Acevedo	Carrilo	Nicte Libertad	5	4	8.00	Ficha
2	21201002	Agular	Zacatenco	Teresa	5	1	2.00	Ficha
3	21201003	Alonso	Solis	Amy Ilse	5	3	6.00	Ficha
4	21201004	Alvarado	Tlatenchi	Teresa	5	3	6.00	Ficha
5	21201005	Aragón	Estudillo	José Juan	5	2	4.00	Ficha
6	21201006	Balbuena	Barreto	Elidey	5	4	8.00	Ficha

Mostrar Gráfica REGRESAR

Figura 4.13 Registro de calificaciones

4.5.14 Pantalla Ficha de calificación

Pantalla que muestra la calificación obtenida de manera individual de una prueba específica.

Escuela Normal Ofl. "Profr. Luis Casarrubias Ibarra"	
Matemáticas y su Enseñanza I	
Fecha: 27/11/2012	
Examen:	Primer momento_24/10/2012
Alumno:	Acevedo Carrilo Nicte Libertad
Semestre:	""
Materia:	Matemáticas y su Enseñanza I
Reactivos:	5
Aciertos:	4
Calificación:	8.00
ACREDITADO	
<input type="button" value="SALIR"/>	
Pregunta	Respuesta
Pregunta: 1	X
Pregunta: 2	1
Pregunta: 3	1
Pregunta: 4	1
Pregunta: 5	1

Figura 4.14 Ficha de Calificación

4.5.15 Pantalla Ficha gráfica resultados por grupo

Pantalla que muestra de manera gráfica las calificaciones de un grupo de alumnos que aplicaron una prueba en común.

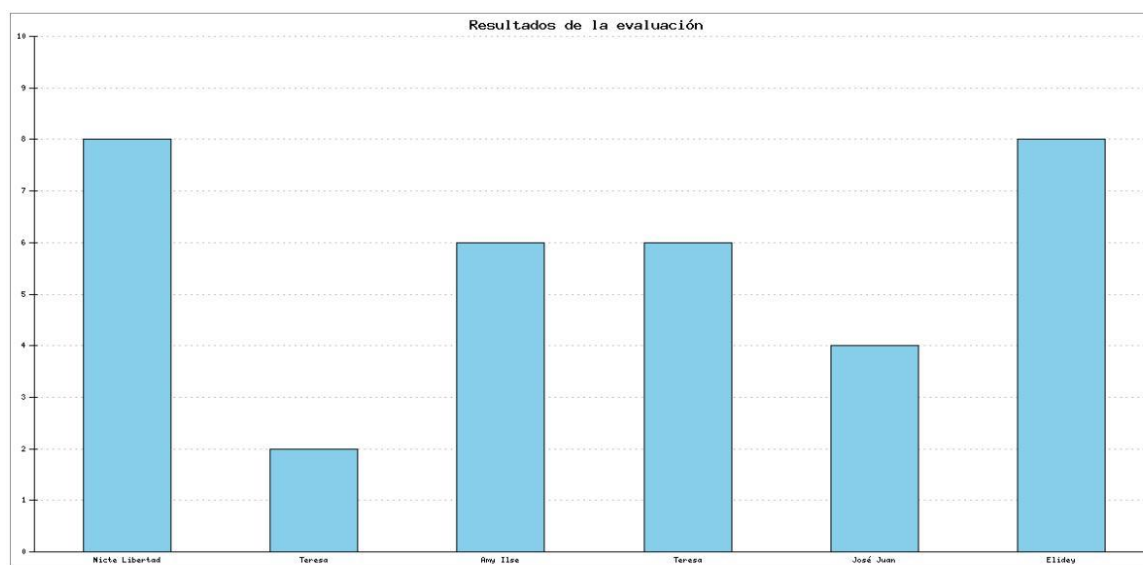


Figura 4.15 Prueba

4.5.16 Pantalla Impresión Resultados

Pantalla que muestra los resultados en un archivo PDF para su impresión.

Secretaría de Educación Pública
Escuela Normal Of. "Profr. Luis Casarubias Ibarra"
Reporte de Calificaciones

Materia:	Matemáticas y su Enseñanza I	Num. de Reactivos:	6
Prueba:	Primer momento_24/10/2012	Aciertos:	4
Fecha de Aplicación:	2012-11-13	Calificación:	8.00
Nombre del Alumno:	Balbuena Barreto Estrey		ACREDITADO

Descripción de la Prueba aplicada

Código pregunta	Pregunta	respuesta
MAT16	1	X
MAT14	2	1
MAT12	3	1
MAT10	4	1
MAT11	6	1

Nombre y Apellido del Usuario: _____

Page 1 of 1

Figura 4.16 Impresión de resultados

4.5.17 Pantalla Prueba

Pantalla que muestra la prueba que debe contestar el alumno.

Secretaría de Educación Pública
Licenciatura en Educación Primaria
Esc. Normal Of. "Profr. Luis Casarubias Ibarra"

Nombre del Alumno: José Juan Aragón Estudillo EXAMEN: Primer momento_24/10/2012

1 **Cuál de las siguientes respuestas no corresponde con las características que debe tener una situación para que sea un problema interesante**

- A) Plantear una meta comprensible para quien la va a realizar.
- B) Permitir aproximaciones a la solución a partir de los conocimientos previos de la persona
- C) Plantear un reto, una dificultad.
- D) ser muy fácil, no debe presentar ninguna dificultad.

2 **Cambio importante en la relación entre conocimientos y problemas que se plantea en el enfoque actual sobre el aprendizaje de las matemáticas en la Escuela Primaria del nuevo Plan y programas de estudios**

- A) Se deben enseñar primeramente los algoritmos para que posteriormente ver situaciones donde los apliquen.

Enviar Salir

Figura 4.17 Prueba

4.6 Estadísticas de Usabilidad

A continuación se muestran los resultados de la encuesta de Usabilidad para el método UsalabCrea, aplicada a 8 docente de la esc. Normal Ofi. "Profr. Luis Casarrubias Ibarra" que probaron el sistema. El cuestionario utilizado puede ser consultado en el anexo A.

En dicha encuesta se evalúan 3 categorías:

- I. Recordar – Visual
- II. Aprender – Organizar contenidos
- III. Eficiencia – Funcionabilidad

Cada pregunta de la encuesta se evalúa de 1 a 5. Correspondiendo el valor de uno a "En Desacuerdo" y el valor de 5 a "De acuerdo".

La siguiente figura muestra la gráfica que representa los valores obtenidos para la categoría Recordar-visual.

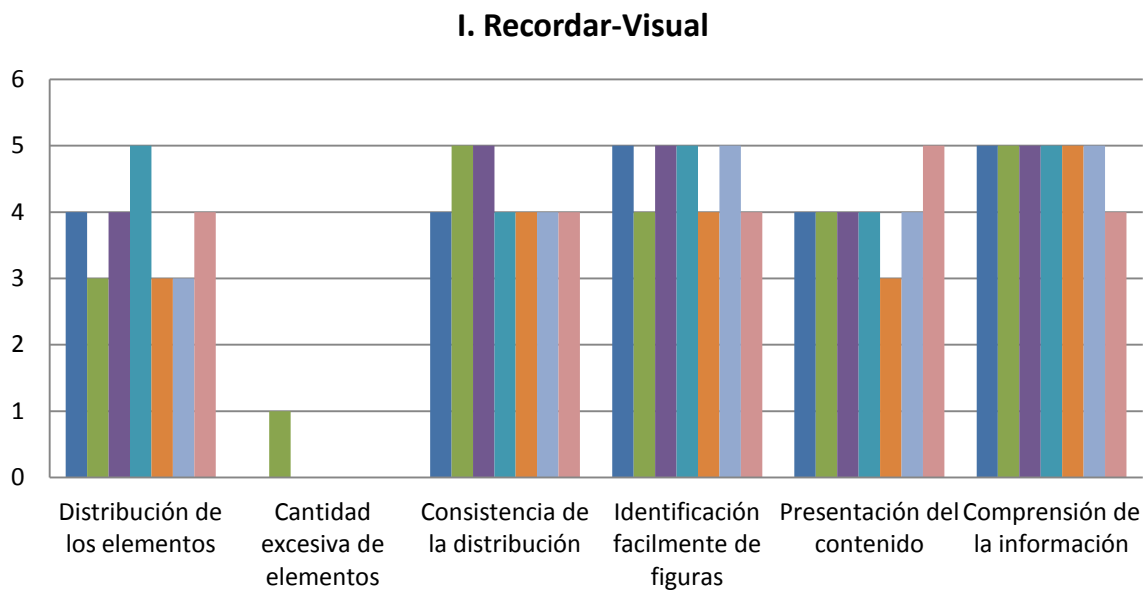


Fig. 4.18 Gráfica Recordar-Visual

Tomando en cuenta la gráfica anterior se obtuvo la satisfactibilidad de los encuestados que se muestra a continuación:

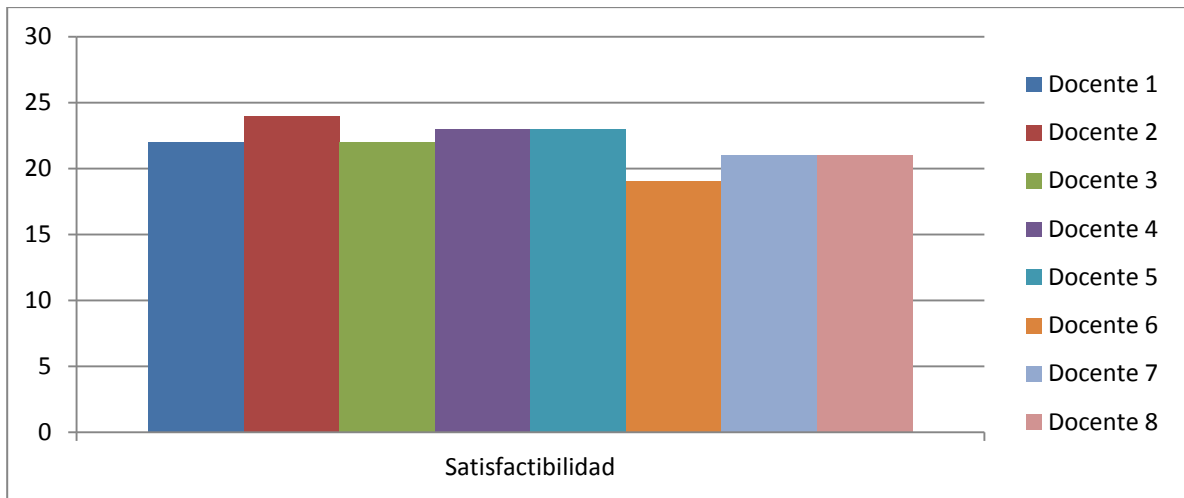


Fig. 4.19 Gráfica de Satisfactibilidad de Recordar-Visual

La media de la gráfica de satisfactibilidad de Recordar-Visual es: 1.61

En figura 4.20 se observa la gráfica que representa los valores obtenidos para la categoría Aprender-organizar contenidos.

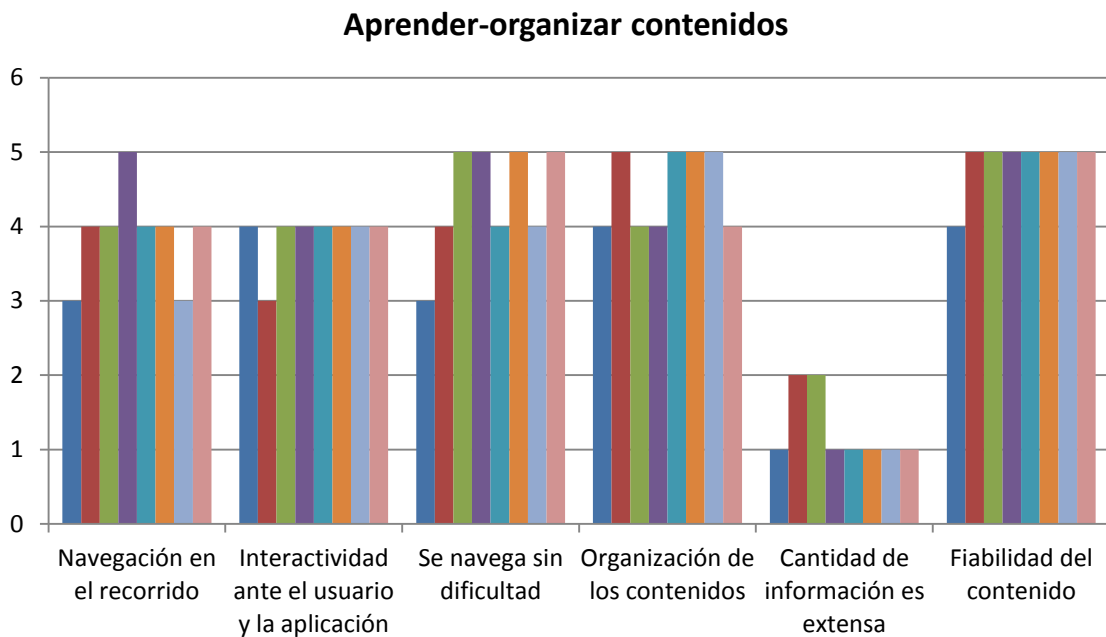


Fig. 4.20 Gráfica Aprender-Organizar contenidos

Teniendo en cuenta la gráfica anterior se obtuvo la satisfactibilidad de los encuestados que se muestra a continuación:

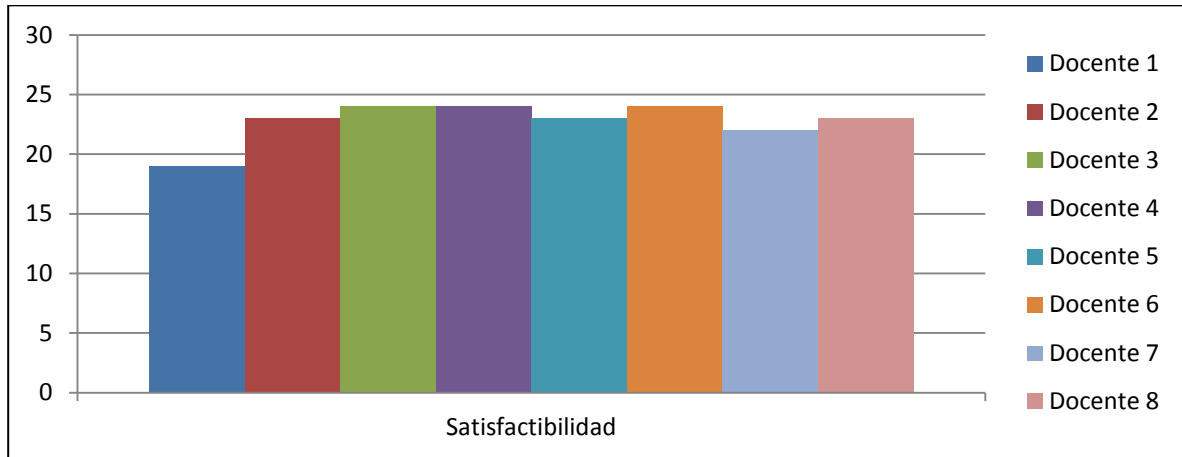


Fig. 4.21 Gráfica de satisfactibilidad Aprender-Organizar contenidos

La media de la gráfica de satisfactibilidad de Aprender - Organizar contenidos: 3.04

En la figura 4.22 se muestra la gráfica que representa los valores obtenidos para la categoría Eficiencia-Funcionabilidad.

III Eficiencia - Funcionabilidad

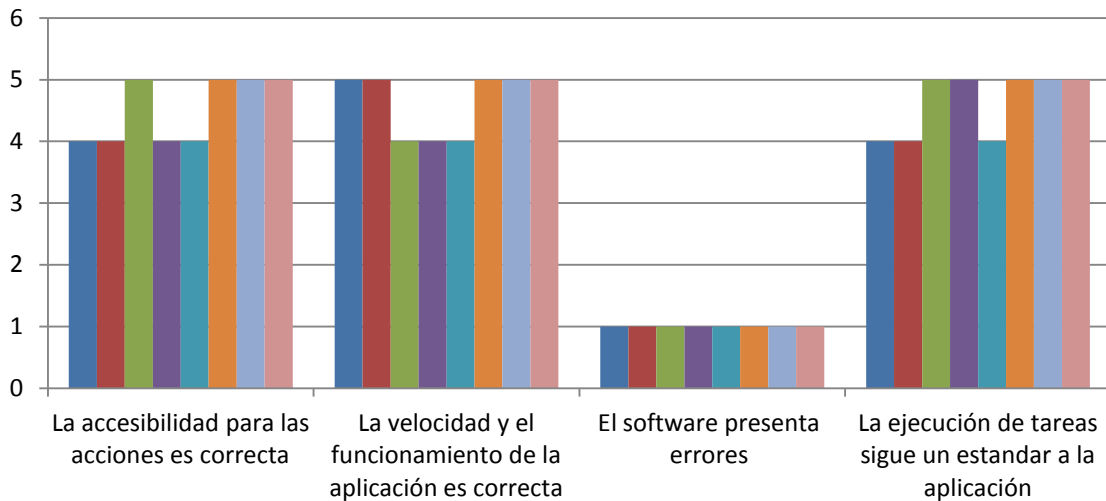


Fig. 4.22 Gráfica Eficiencia-Funcionabilidad

Tomando en cuenta la gráfica anterior, se obtuvo la satisfactibilidad de los encuestados que se muestra a continuación:

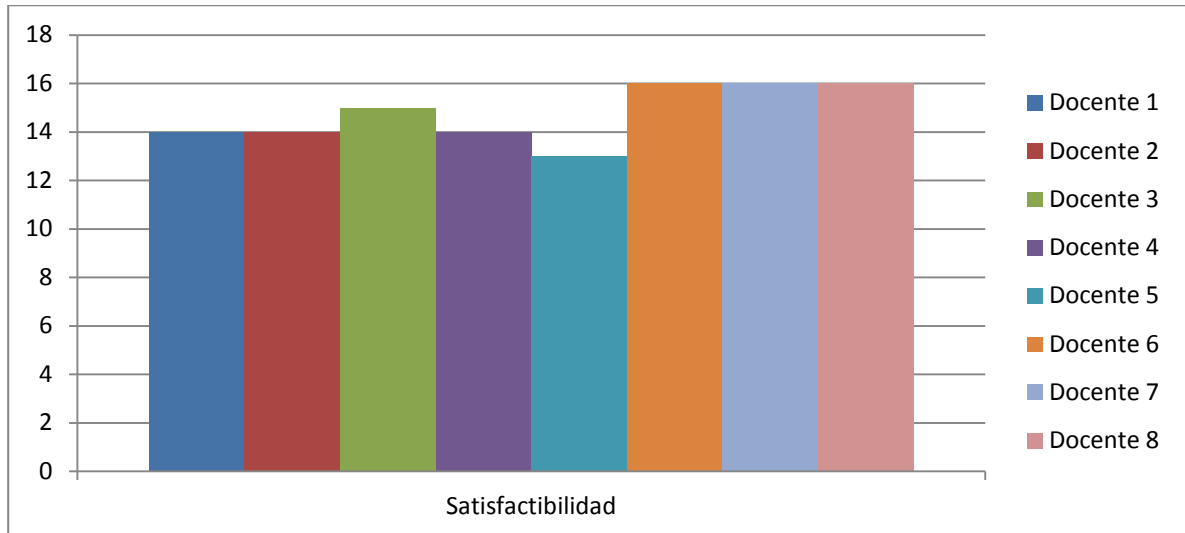


Fig. 4.23 Gráfica de satisfactibilidad Eficiencia-Funcionabilidad

La media de la gráfica de satisfactibilidad Eficiencia - Funcionabilidad: 2.41

CONCLUSIONES

- El trabajo es resultado de aplicar la metodología de diseño de desarrollo de software y el modelo de cascada e implementado con los lenguajes de programación dinámica de sitios web como son PHP, HTML, JavaScript y Ajax. Además se utilizó como gestor de bases de datos MySQL para el almacenamiento de la información.
- El sistema se piloteó con docentes y alumnos para encontrar posibles errores y lograr que sea eficiente y funcional, sin elementos visuales excesivos, con una interfaz amena y sin saturar las pantallas de información.
- Se diseñaron las interfaces gráficas con los colores propios de la institución, dándole mayor presentación y buena interactividad entre el sistema y el usuario.
- El sistema cumple con el objetivo principal de proporcionar a la escuela normal de una herramienta que permite la creación de un banco de reactivos, la aplicación y calificación de pruebas objetivas y consulta de calificaciones; de manera rápida y económica. Además se lograron las siguientes funcionalidades:
 - Gestionar los distintos usuarios del sistema
 - Gestionar preguntas del banco de reactivos
 - Gestionar pruebas objetivas
 - Gestionar la asignación de pruebas objetivas
 - Generar record de los sustentantes
 - Generar en un archivo PDF el resultado obtenido por los sustentantes
 - Mostrar de manera gráfica los resultados obtenidos, por grupo o por subtema.

PERSPECTIVAS

A continuación se proponen las siguientes perspectivas para el mejoramiento del sistema tomando en cuenta las indicaciones de las encuestas realizadas a los usuarios.

- Reducir las pantallas que se muestran, mejorando así la navegación y la interfaz con el usuario.
- Permitir la elaboración y aplicación de pruebas con preguntas de tipo relación
- Incluir imágenes en las preguntas y respuestas
- Crear un listado de calificaciones para que puedan ser abiertos en Microsoft Excel
- Mejorar el diseño de las pantallas

REFERENCIAS BIBLIOGRÁFICAS

- [1] Roger S. Pressman. "Ingeniería Del Software – V edición". Editorial: McGraw-Hill
- [2] I.Sommerville. "Ingeniería del Software" – VII edición " Editorial: Addison Wesley
- [3] <https://sites.google.com/site/ingenieria3a/ingenieria-de-software>
- [4] http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software
- [5] <http://taziv.blogspot.mx/2007/03/ciclos-de-vida-del-software.html>
- [6] http://es.wikipedia.org/wiki/Desarrollo_en_espiral
- [7] http://www.ecured.cu/index.php/Modelo_de_Prototipos
- [8] <http://www.conocimientosweb.net/portal/article2976.html>
- [9] http://www.ecured.cu/index.php/Metodolog%C3%ADa_de_desarrollo_iterativo_y_creciente
- [10] Craig Larman "UML Y PATRONES Introducción al análisis y diseño orientado a objetos".
Editorial: Prentice Hall, Addison Wesley Longman.
- [11] Joseph Schmuller. "Aprendiendo UML en 24 horas". Editorial: Prentice Hall
- [12] <http://www.monografias.com/trabajos94/modelado-sistemas-uml/modelado-sistemas-uml.shtml>
- [13] Abraham Silberschatz, Henry F. Korth "Fundamentos de Bases de Datos –IV edición"
Editorial: McGraw-Hill
- [14] Alma Delia Ambrosio Vázquez "Notas Del curso 'Diseño de Bases de Datos' Diplomado
en Tecnologías de la Información, otoño 2010. FCC, BUAP

[15] <http://ar.emagister.com/cursos-gratis/normalizacion-bases-datos-cursos-2514356.htm>

[16] M.C. Nancy Paola Cira Pérez. “Notas de Bases de Datos”.

[17] http://es.wikipedia.org/wiki/Servidor_HTTP_Apache

[18] <http://www.buenastareas.com/ensayos/Php-y-Asp/1826288.html>

[19] José López Quijado. “Domine PHP y MySQL: Programación dinámica en el lado del servidor”. Editorial: Alfaomega Ra-Ma. M. Mayo 2007

[20] <http://www.emagister.com/curso-mysql-php/introduccion-bases-datos-mysql>

[21] <http://es.wikipedia.org/wiki/AJAX>

[22] Patricia Frola. “Competencias Docentes para la Evaluación: Diseño de reactivos para evaluar el aprendizaje”. Editorial Trillas, 2008

[23] <http://www.dgespe.sep.gob.mx/dsi/egc/informes>

[24] Archundia E. (2009). Tesis doctoral: Metodología para propiciar el aprendizaje de la programación creativa APROGC. Universidad Popular Autónoma del Estado de Puebla, México. (pp.101)

[25] http://es.wikibooks.org/wiki/Fundamentos_de_programaci%C3%B3n/Herramientas_de_desarrollo

GLOSARIO

Acta de nacimiento del reactivo: Formato que sirve para registrar, identificar y ubicar fácilmente al reactivo y a su elaborador; ayuda a mantener un estricto control de los reactivos, y por ende la calidad del proceso de validación y calibración.

AJAX: Acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas

Aleatorio: Que depende de la suerte o el azar.

Aplicación del Examen: Proceso mediante el cual un conjunto de estudiantes responde un examen con el propósito de obtener un indicador de su rendimiento.

Banco de Reactivos: Conjunto de reactivos elaborados para un examen.

CENEVAL: Centro Nacional de Evaluación

EGC-LEPRI: Examen General de Conocimientos de la Licenciatura en Educación Primaria

Evaluación: Proceso destinado a obtener información sobre un fenómeno, sujeto u objeto de evaluación; emitir juicios de valor al respecto y, con base en ellos, tomar decisiones.

Examen de Oposición: Examen que presentan los egresados de alguna escuela Normal para el ingreso al servicio docente.

EXI-LEPRI: Examen Intermedio de Conocimientos de la Licenciatura en Educación Primaria

Formatos del reactivo: Formatos en que se pueden presentar los reactivos. Los 5 principales son: Cuestionamiento directo, Tipo canevá o de completar, ordenamiento y selección de elementos, relación de columnas o apareamiento y opción alterna, falso-verdadero.

HTML: Siglas de HyperText Markup Language (lenguaje de marcado de hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web.

Javascript: Lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Perfil de egreso: Es la delimitación de las competencias que un estudiante debe evidenciar y sustentar al egresar de un nivel educativo, ciclo o grado académico.

PHP: Es un acrónimo recursivo que significa *PHP Hypertext Pre-processor*. Lenguaje de programación usado generalmente en la creación de contenidos para sitios web. Es un lenguaje interpretado especialmente usado para crear contenido dinámico web y aplicaciones para servidores

Programación dinámica: Es un método para reducir el tiempo de ejecución de un algoritmo mediante la utilización de subproblemas superpuestos y subestructuras óptimas.

Pruebas objetivas: Instrumentos de evaluación conocidos también como exámenes de opción múltiple o de respuesta estructurada, ya que el sustentante debe elegir solamente la opción que incluya la respuesta correcta; no se requiere rescribir o redactar la respuesta ni desarrollar temas abiertos.

Reactivo: Es una pregunta, problema o cuestionamiento, a partir del cual se puede inferir la ejecución, desempeño o manifestación de una actitud o habilidad del sustentante, que constituye la materia prima del instrumento de medición.

Reactivo de completamiento o de canevá: Formato de reactivo que consiste en una serie de oraciones o frases incompletas, donde el examinado debe elegir la opción que sustituya las líneas vacías con palabras correctas que completen dicha frase o enunciado.

Reactivo de opción múltiple: Enunciado que presenta la situación, caso o problema planteado, explícita o implícitamente en una pregunta, afirmación o enunciado incompleto y que ofrece cuatro o cinco opciones que se entienden como posibles respuestas.

Reactivo simple: Se caracteriza por plantear una sola pregunta, situación o problema, a través de un enunciado que puede ser formulado en forma interrogativa, imperativa, afirmativa y de manera excepcional en forma negativa.

SEP: Secretaría de Educación Pública

SGBD: Siglas utilizadas para Sistema Gestor de Bases de Datos. Son un tipo de software muy específico, dedicado a servir de interfaz, entre la base de datos, el usuario y las aplicaciones que la utilizan.

MySQL: Es un sistema de gestión de bases de datos, multiusuario, multiplataforma y de código abierto.

TIC: Tecnologías de Información y Comunicaciones.

UML: Siglas utilizadas para Lenguaje Unificado de Modelado. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

ANEXO A

Cuestionario UsalabCrea

Instrumento No. 1

TEST de Usabilidad para el Método UsalabCrea ^[24]

Fecha _____

El siguiente cuestionario tiene la finalidad de mostrar la prueba de usabilidad relacionada con la creatividad para medir el grado de satisfacción (recordar, aprender y de eficiencia) de los usuarios respecto del producto de software. Tache con una X el cuadro que indique en su opinión el grado de satisfacción (en desacuerdo hasta acuerdo) a las observaciones que a continuación se presentan.

El siguiente instrumento mide el grado de satisfacción del producto de software denominado _____.

I. Recordar-visual

1. Las distribuciones de los elementos de la aplicación del uso de los mecanismos de control como botones, barras y menús se consideran bien organizadas.

En desacuerdo De acuerdo

2. La cantidad de elementos visuales que utilizan en la aplicación es excesiva

En desacuerdo De acuerdo

3. La consistencia de la distribución de los elementos estructurales se mantiene constante a lo largo de la aplicación.

En desacuerdo De acuerdo

4. Se identifica fácilmente las figuras, tablas, los hipertextos, las zonas activas y el tipo de acción que se debe ejecutar.

En desacuerdo De acuerdo

5. La presentación del contenido en cuanto a uso del color, el tamaño de la fuente, el significado de las figuras se considera buena.

En desacuerdo De acuerdo

6. La comprensión de la información que se presenta en la aplicación es fácil de entender y por lo tanto de recordar.

En desacuerdo De acuerdo

II. Aprender-organizar contenidos

1. La navegación en el recorrido que se hace por el contenido de la aplicación es entendible.

En desacuerdo De acuerdo

2. Se logró la interactividad entre el usuario y la aplicación

En desacuerdo De acuerdo

3. Los procedimientos de navegación por la aplicación o ejecución de las tareas asignadas se realizan sin dificultad.

En desacuerdo De acuerdo

4. La organización de los contenidos de los textos e imágenes en la aplicación la considera adecuada.

En desacuerdo De acuerdo

5. La cantidad de información que se presenta en la aplicación es demasiado extensa.

En desacuerdo De acuerdo

6. La fiabilidad del contenido se presenta por la carencia de errores en la información que se presenten en la aplicación.

En desacuerdo De acuerdo

III. Eficiencia-funcionabilidad

1. La accesibilidad para las acciones que solicita la aplicación es adecuada

En desacuerdo De acuerdo

2. La velocidad y el funcionamiento de la aplicación considerando el tipo de tarea que se exige es correcta.

En desacuerdo De acuerdo

3. El software presenta un número significativo de errores durante su operación.

En desacuerdo De acuerdo

4. La ejecución de tareas sigue un estándar a la aplicación.

En desacuerdo De acuerdo

Opinión general de la aplicación

Gracias por su atención

Autor:

Archundia E. (2009). Tesis doctoral: Metodología para propiciar el aprendizaje de la programación creativa APROGC. Universidad Popular Autónoma del Estado de Puebla, México. (pp.101)