UNIVERSIDAD DE PINAR DEL RÍO
“Hermanos Saíz Montes de Oca”

“Sistema para el Control de las Incidencias de los Policlínicos Facultad de Ciencias Médicas. SISCOIN”.

Tesis presentada en opción al Título Académico de Máster en Nuevas Tecnologías para la Educación.

Autor: Ing. Ernesto Pérez Rojas

Tutores: Msc. Iván Valdés Rodríguez
Msc. Oliver Milan Telleria

Pinar del Río, 2010

DEDICATORIA

A mis padres, hermanos, esposa e hijos, familiares y amigos

AGRADECIMIENTOS

Sería imposible, aun cuando lo intentásemos, poner en tan poco espacio a todos aquellos que de una forma u otra contribuyeron a la realización de este trabajo. Es por ello que preferimos mencionar solamente a los que han puesto su granito de arena en este proyecto.

A mis padres y hermanos, esposa e hijos, sin su comprensión no hubiera llegado al final. A mis padres por saber guiarme y saber educarme con todas las mejores ideas del mundo. A mi esposa e hijos por su apoyo en todo momento y por el tiempo que deje de dedicarles.

A todos los profesores del departamento de informática por su ayuda incondicional, que nos han apoyado en todo momento a lo largo de estos seis años.

A los Ingenieros jefe de órgano Juan Carlos Medina Mas y jefa de grupo Marlín Rojas Álvarez por su apoyo y preocupación.

A mis compañeros de trabajo (Neysi, Roxana, Dailyn, Lien, Yilian, Alexander, Janoi, Félix, Abreu, Montesino) y todos los que han logrado que llegue este momento.

“Sistema para el Control de las Incidencias de los Policlínicos Facultad de Ciencias Médicas. SISCOIN”.

Ing. Ernesto Pérez Rojas

Jefatura Provincial del MININT Pinar del Río

RESUMEN

El presente trabajo se desarrolló en el Órgano de Informática y Comunicaciones atendiendo a una problemática del Órgano de Dirección, teniendo como función principal la supervisión de los Policlínicos Facultad de Ciencias Médicas en el municipio de Sandino y utilizar el sistema con la información acumulada en la SUM del MININT para llevar a cabo la preparación de los estudiantes a través de situaciones problémicas a resolver accediendo a las bases de datos.

El control de esta actividad se realizaba de forma manual, lo que dificulta la efectividad de la misma, ya que la información no se explota de forma óptima tanto en el trabajo operativo, como en la toma de decisiones sobre incidencias, enfermedades, vías de acceso, problemas constructivos, etc.

El portal de enfrentamiento en el cual se encuentra el SISCOIN con el fin de llevar una mejor preparación en la formación delos futuros oficiales operativos en la SUM MININT fue creado en Delphi mediante el cual se accede a través de vínculos a los distintos sistemas automatizados que lo integran.

El sistema **SISCOIN** fue desarrollado en tecnología ASP con el lenguaje de programación VBScript utilizando la herramienta DreamWeaver de Macromedia, el mismo se implementó en el servidor nativo de ASP (Página de Servidor Activa) el IIS (Internet Information Server) de Microsoft, esto permite hacer las transacciones hacia el servidor de datos de forma rápida y sencilla. Hay que tener en cuenta que existen otras compañías que proveen soporte para ASP en servidores fuera de IIS.

En la realización de nuestro sistema se hizo uso de la Metodología RUP (Rational Unified Process). Para el diseño de la base de datos se utilizó la herramienta CASE Rational Rose Enterprise Edition y como motor de bases de datos para su implementación se utilizó el ORACLE 11G ya que es el que explota la institución como política de la dirección, y la ayuda la desarrollamos con RoboHelp.

Como resultado de la investigación se logra la supervisión de los Policlínicos Facultad de Ciencia médicas y se integra el sistema en la preparación informática operativa en la SUM MININT.

LENGUAJE ESTRUCTURADO DE CONSULTA (SQL): Es un sub lenguaje que tiene como propósito ser una interface para las bases de datos, donde todas las sentencias son instrucciones de la base da datos.

TABLESPACE: Unidad lógica en la que se crean los objetos de una base de datos. Una base de datos puede tener varios tablespace.

TRIGGERS: Unidad de programa que se ejecuta automáticamente cuando ocurre el evento al cual está asociado

VISTAS: Tablas virtuales o consultas almacenadas que no ocupan espacio de datos, mejoran el rendimiento en el acceso a los datos.

Pág.

INTRODUCCIÓN 1

CAPÍTULO I. BASES PRELIMINARES 5

1.1 Introducción 5

1.2 Caracterización del entorno 5

1.3 Justificación de la solución del problema con el empleo dela NTIC 7
1.3.1 Modelo del Dominio *7*

1.4 Análisis de factibilidad 8
1.4.1 Estimación de costos del desarrollo del sistema 8
1.4.2 Análisis de beneficios tangibles e intangibles 13

1.5 Conclusiones 14

CAPÍTULO II. TENDENCIAS Y TECNOLOGÍAS ACTULES 15

2.1 Introducción 15

2.2 Uso de las TICS en la educación. 15

2.3 Portal de Enfrentamiento. 16

2.4 Herramientas, lenguajes y metodologías usadas para el desarrollo del sistema. 16
2.4.1 Tecnologías Web 16
2.4.2 Servidores Web 17
2.4.3 Internet Information Server 17
2.4.4 ¿Por qué utilizar ISS? 18
2.4.5 Características fundamentales de Dreamweaver 18
2.4.6 Definición de ASP 19
2.4.7 Aplicaciones ASP 19
2.4.8 Objetos integrados en ASP *20*
2.4.9 ASP en comparación con tecnologías actuales 21
2.4.10 Ventajas de usar ASP 22

2.5 ¿Por que utilizar una Base de Datos? 22
2.5.1 Tipos de Base de Datos 23
2.5.2 Gestores de Base de Datos 23
2.5.3 ¿Por qué utilizar Oracle 11G como Gestor de Bases de Datos? 24
2.5.4 Principales posibilidades que implementa Oracle Server 24
2.5.5 Elementos básicos de Oracle. Ficheros de Datos y Tablespaces. 25
2.5.6 Objetos que conforman una Base de Datos Oracle 26

2.5.7 Administración y seguridad 27
2.5.8 Definición de Usuarios 28
2.5.9 Privilegios 29
2.5.10 Roles 30
2.5.11 Reglas de integridad 30
2.5.12 Restricciones de integridad Declarativas 31
2.5.13 Tipos de restricciones 32
2.5.14 Integridad Referencial 33

2.6 Metodologías. Proceso Unificado de Desarrollo (RUP). 34
2.6.1 Extreme Programing (XP) 35
2.6.2 Microsoft Solution Framework (MSF) 36
2.6.3 ¿Por qué se utilizó Rational Unified Process (RUP)? 36

2.7 Herramientas CASE 36
2.7.2 Lenguaje de Modelamiento Unificado (UML). 37

2.8 Conclusiones 38

CAPÍTULO III. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA 38

3.1 Introducción 38

3.2 Modelo de Diseño 38
3.2.1 Patrones 38
3.2.2 Diseño de la Interfaz de la Aplicación 38
3.2.3 Estándares de la interfaz 39
3.2.4 Requerimientos Funcionales 39
3.2.5 Requerimientos No Funcionales 40

3.3 Modelo del sistema 42
3.3.1 Actores del sistema 42
3.3.2 Modelo de casos de uso del Sistema 42

3.4 Descripción de los casos de uso del sistema. 43
3.4.1 Gestión de la Seguridad *43*

3.5 Diagramas de Clases Web 56

3.6 Clases Persistentes 56

3.7 Modelo de Datos 56

3.8 Mapa de Navegación 56

3.9 Modelo de Despliegue 57
3.9.1 Características principales de los servidores *59*

3.10 Resultados de la consulta a especialistas para la comprobación del grado de validez del sistema que se proponen. 60

3.11 Conclusiones 60

CONCLUSIONES 61

RECOMENDACIONES 62

REFERENCIAS BIBLIOGRÁFICAS 63

BIBLIOGRAFÍA 64

ANEXOS 67

INTRODUCCIÓN

Con los problemas existentes en la sociedad de una forma u otra todos estamos involucrados en la lucha por un mundo mejor, donde el Ministerio del Interior juega un papel importante para lograr y mantener la tranquilidad ciudadana que tan valiosa es para una sociedad justa como la nuestra, así como en el mejoramiento del control de todos los recursos que afectan la economía del país y en otras vertientes encaminadas a lograr la integración de las Américas.

Uno de los acuerdos del ALBA es la creación de los Policlínicos Facultad de Ciencias Médicas en el municipio Sandino, donde se formarán profesionales de la salud de los distintos países de la región y otras partes del mundo, abriendo esto un camino a las relaciones económicas y diplomáticas de estos países con nuestra isla, aportando los nuestros, el potencial humano. Los futuros profesionales se alejan de su tierra natal así como de sus familiares queridos por un período de tiempo prolongado y la Revolución los acoge en su seno como sus hijos, responsabilizándose con ellos en cuanto a la creación de las condiciones mínimas indispensables en estos centros.

Por tanto la jefatura del MININT en nuestra provincia de conjunto con el Órgano de Informática y Comunicaciones, se dio la tarea de elaborar un sistema en ambiente Web para controlar las incidencias que puedan ocurrir en las distintas escuelas que afecten directamente al estudiantado. Dicho sistema se nombra SISCOIN (Sistema para el Control de Incidencias), el cual debe contribuir a perfeccionar el nuevo modelo de estudios.

No todos los oficiales de nuestro Ministerio del Interior están preparados adecuadamente para darle solución a determinados problemas o esclarecer situaciones en el menor tiempo posible utilizando las NTIC por lo que la jefatura provincial del MININT decidió crear un portal de enfrentamiento con determinados sistemas en el cual se encuentra el SISCOIN para llevar a cabo la preparación de los oficiales operativos en la SUM MININT.

Esta decisión se toma a partir de la siguiente **situación problémica:** Ineficiencia en el control de las incidencias de los Policlínicos Facultad de Ciencias Médicas y falta de preparación de

los oficiales operativos. Lo que incurre en la preparación y formación de los futuros profesionales de la salud. De ahí que el **problema científico** de este trabajo se identifica como: inexistencia de un software que permita controlar y a su vez brindar información acerca de las incidencias que ocurren en dichos centros para la toma de decisiones ante determinadas situaciones.

En correspondencia con el problema el **objeto de estudio** lo constituye el control de las incidencias ocurridas en los Policlínicos Facultad de Ciencias Médicas. El **campo de acción** lleva implícito el perfeccionamiento del control de las incidencias mediante el sistema automatizado.

Se pretende como **objetivo general** del presente trabajo:

Desarrollar un sistema automatizado que nos permita almacenar, actualizar y recuperar toda la información estructurada referente a las incidencias ocurridas en los Policlínicos Facultad de Ciencias Médicas.

Del objetivo general de este trabajo se derivan los siguientes **objetivos específicos**.

1. Determinar los elementos que intervienen en el proceso de control de las incidencias.
2. Diseñar un sistema que permita una sencilla navegación por los distintos módulos del sistema.
3. Implementar el sistema basado en las redes de comunicaciones del MININT.

Las tareas a desarrollar para dar cumplimiento a estos objetivos son:

1. Indagar si existe alguna aplicación afín con la propuesta.
2. Definir las características que tendrá el software resultante.
3. Valorar la capacidad disponible de los servidores.
4. Analizar la velocidad de conexión de los clientes con los servidores.
5. Diseñar la base de datos.
6. Diseñar el sistema.
7. Brindar herramientas eficientes que permitan gestionar la información.
8. Establecer mecanismos de seguridad, control y protección a nivel de los usuarios del sistema y la base de datos.

En cuanto a los **antecedentes** sobre la temática de la Supervisión de los Policlínicos Facultad de Ciencias Médicas se realizó una investigación con ayuda del Órgano de Dirección, sobre la existencia o no, de un sistema automatizado en el resto de las provincias del país que reuniera los requisitos para ser implementado en nuestro entorno, ya que se controla un volumen de información de forma manual, siendo insuficiente su control, lo que impide dar respuesta de manera eficiente y oportuna a determinadas situaciones. Debido a la inexistencia de un sistema de este tipo de control surge la necesidad de automatizar la información y a su vez utilizarlo en la preparación de los oficiales operativos durante su formación o en cursos para que exploten de manera eficaz las tecnologías de informática y comunicaciones.

Su **valor práctico** está determinado, a través, de la elaboración de un software donde pretendemos poner en manos de los oficiales una herramienta que ayude a perfeccionar su labor en la toma de decisiones, y con ello mantener la tranquilidad y el orden en dichos centros escolares.

El presente documento está estructurado en tres capítulos.

Capítulo 1: En el capítulo se hace una caracterización del entorno, se justifica como se soluciona el problema mediante el empleo de las TICS, se representan las entidades más importantes mediante el modelo del dominio y se hace el análisis de factibilidad; estimación de costos y beneficios del desarrollo del sistema que lleva implícito los recursos humanos y tecnológicos a través del COCOMO.

Capítulo 2: Este capítulo trata el vínculo de la aplicación en el proceso educacional, su importancia en el MININT, el beneficio que le aporta al centro de instrucción donde se encuentra encabada la SUM MININT, además, aborda el estado del arte de la tecnología donde se justifica las tecnologías que se utilizan como es el caso de ORACLE (Sistema Gestor de Base de Datos) haciendo referencia a su arquitectura, seguridad, entre otros puntos a tener en cuenta a la hora de implementar una base de datos sobre ORACLE. También se explica el funcionamiento de otras herramientas utilizadas en la aplicación como son Dreamweaver, el empleo de la tecnología ASP y Visual Basic Script como lenguaje de programación.

.

Capítulo 3: En el desarrollo de este capítulo se explicarán aspectos relacionados con el diseño e implementación del sistema dentro de los que se encuentran diseño interfaz – usuario, así como la especificación de los requisitos funcionales y no funcionales. En el modelo del sistema se relacionan los actores del sistema y casos de usos. Se describen explícitamente los casos de usos más importantes del sistema. Se describe el funcionamiento del software mediante la navegación por los distintos módulos que componen el sistema basándose en el modelo de datos, diseño de interfaz de la aplicación, mapa de navegación y modelo de despliegue.

CAPÍTULO I. BASES PRELIMINARES

1.1 Introducción

En el desarrollo de este capítulo se explicará el surgimiento del Sistema para el Control de las Incidencias en los Policlínicos Facultad de Ciencias Médicas mediante la caracterización del entorno, la solución del problema expuesto mediantes las TICS.

Además se aborda el modelo del dominio con todos los diagramas que lo componen y se realiza la estimación de costos de desarrollo del sistema con el uso de la herramienta COCOMO.

1.2 Caracterización del entorno

El proyecto de Policlínico Universitario es una iniciativa revolucionaria acorde con la pedagogía actual en el mundo. Las universidades persiguen la esencialidad del conocimiento y la integración de las asignaturas. La modalidad de los estudios en los municipios es una experiencia joven en la que se han dado pasos de avances acercando al estudiante al escenario de trabajo y requiere de una esmerada atención del tutor, reafirmándose el principio de aprender haciendo.

El proyecto es amplio y desde el principio el Comandante en Jefe Fidel Castro planteó que se hiciera una investigación vinculada al proyecto, con el objetivo de detectar fallas y corregirlas ya que es la vía de perfeccionar el modelo.

La inauguración de los Policlínicos Facultades de Ciencias Médicas y la matrícula de los estudiantes extranjeros para llevar a cabo el programa de formación del nuevo médico y la creación de las SUM en el MININT, entre otras tareas dio origen al sistema automatizado SISCOIN para controlar las incidencias que ocurren en los centros de estudios o fuera de estos, siempre y cuando existan alumnos o trabajadores involucrados como autores, coautores, cómplices o victimas.

Se considera incidencia todo lo que influya negativamente sobre los estudiantes, un accidente de tránsito es una incidencia donde los lesionados (involucrados) son víctimas a los que hay que darle seguimiento al estado de salud, cómo evolucionan, en qué hospital se atienden y en

dependencia del tiempo que van a estar alejado de la docencia, trazar una estrategia. Otro ejemplo puede ser los problemas constructivos de un centro, como puede ser problemas con instalaciones hidrosanitarias que afecta las condiciones de vida e impide un mejor aprovechamiento de los estudiantes, además, esta situación provoca que los estudiantes con posibilidades económicas hagan vida nocturna en casas particulares arrendadas, lo que conlleva al surgimiento de incidencias aún más graves ya que se van fuera del radio de acción de la dirección del centro.

Los ejemplos de incidencias son múltiples, estos van desde la pérdida de un objeto personal, hasta la solicitud propia de traslado o baja del centro de estudios por determinadas causas.

Anteriormente el control de las incidencias las tenía en expedientes archivados el oficial que las tributaba en dependencia del órgano al que pertenecía, lo que impedía que el cúmulo de información de los diferentes factores que intervenían se concentrara para su mejor explotación. Con la implementación del sistema en el cual se almacenan las informaciones en una base de datos relacional, se resuelve esta disyuntiva.

El control de las incidencias en estos PFCM es de vital importancia porque se tiene una estadística de los alumnos de las nacionalidades o países que más incurren por determinados parámetros: período de tiempo, clasificación de las incidencias, órganos que las tributan, centro de estudios al que pertenecen, además se tiene el chequeo de los estudiantes con el grado de participación en las mismas, lo que permite hacer trabajos diferenciados de conjunto con la dirección de los centros.

En los epígrafes siguientes se hace referencia a las herramientas utilizadas en la confección del sistema (SISCOIN) y se explica la necesidad de utilizar una base de datos, así como las posibilidades y características del SGBD utilizado.

1.3 Justificación de la solución del problema con el empleo dela NTIC

En la actualidad se vive inmerso en una sociedad donde el recurso vital y valioso en muchas de las actividades de la vida diaria es la información.

La sociedad actual es producto de una evolución que tuvo la sociedad industrial que producía bienes de consumo a una sociedad de la información, en donde el 80 % de los trabajadores, de una u otra manera trabajan con este elemento intangible.

El hombre vive inmerso y bombardeado por la información, en los medios de comunicación, como TV, Radio, Internet, diarios, multimedia, etc. Pero esto no significa que tenga más conocimiento, ya que se le hace difícil absorber tanta información que le llega al individuo por estos medios y transformarlo en conocimiento.

Mediante la red interna del MININT se logra una adecuada explotación del software resultante puesto que el mismo esta diseñado en ambiente web por lo que se obtiene información en tiempo real, por una parte los oficiales encargados de mantener actualizada la información lo hacen desde su puesto de trabajo en el municipio Sandino. Los estudiantes de la SUM MININT interactúan con el mismo desde el centro de instrucción utilizándolo como medio de enseñanza para darle solución a las situaciones problémica que los profesores le plantean, logrando con ello una mejor preparación profesional a través de las TICs.

1.3.1 Modelo del Dominio

Para la identificación de requisitos y la comprensión del contexto, la metodología RUP propone dos procesos fundamentales la Modelación del Negocio y la Modelación del Dominio.

El Modelo del Dominio es un diagrama utilizado para comprender, capturar y describir las clases más importantes dentro del contexto del sistema **[5].** No es posible identificar un negocio en el sistema, por lo que es necesario realizar un Modelo del Dominio apoyado en el Glosario de Términos. Mediante el Modelo del Dominio se logrará comprender el sistema que se propone.

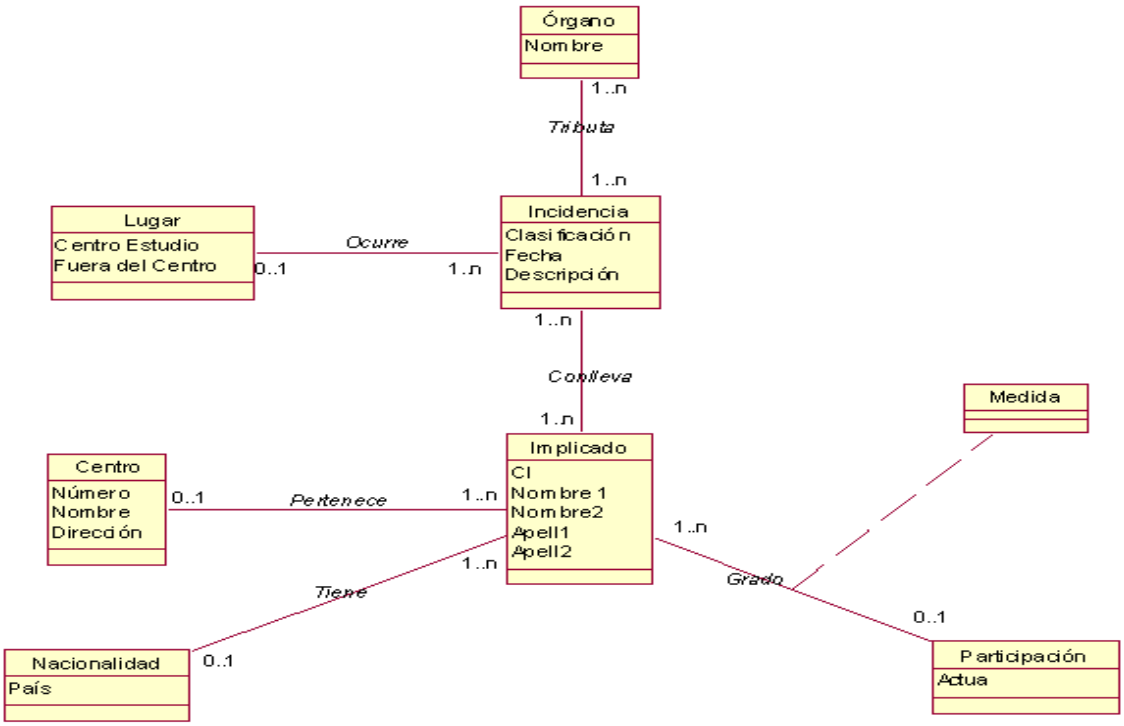


Figura 1.3 Modelo del dominio

En el Modelo del Dominio que se muestra en la figura 1.3 se representan las entidades más importantes que forman parte del proceso de supervisión de los PFCM. Cada incidencia que se gestiona tiene una serie de conceptos asociados a través de enlaces.

1.4 Análisis de factibilidad

1.4.1 Estimación de costos del desarrollo del sistema

En la actualidad existen diversos modelos para estimar los costos de los proyectos de software. Uno de los más aceptados es el modelo COCOMO II (del inglés Constructive Cost Model). El modelo COCOMO fue originalmente publicado en software Engineering Economics por Barry Boehm en 1981 y la versión COCOMO II del 2000 actualiza y extiende este estándar de estimación.

Este modelo expresa el esfuerzo de desarrollo en términos de Personas Mes (cantidad de tiempo que una persona dedica a trabajar sobre el proyecto de desarrollo durante un mes). Finalmente se hace una estimación del costo monetario del proyecto teniendo en cuenta el

tiempo estimado de desarrollo, la cantidad de personas involucradas y el salario de estas personas.

Estimación del Costo

Para realizar el cálculo de los costos de desarrollo del sistema se deben obtener primero las instrucciones fuentes. Analizándose para esto las cantidades de entradas, salidas, peticiones, archivos lógicos e interfaces externas preliminares que tiene el sistema.

Obtención de los puntos de función (UFP).

Para las entradas externas, salidas externas y consultas externas el nivel de complejidad se determina por la cantidad de archivos referenciados y los elementos de datos, mientras que para los archivos lógicos internos y los archivos de interface externos se determina por los tipos de registros y los elementos de datos **[6]**.

Los valores numéricos que se le asignan a cada complejidad para cada una de las transacciones, se muestran en la siguiente tabla.

Tabla 1.1 Asignación de valores numéricos.

Archivos Referenciados	**Valores**		
	Entradas Externas	**Salidas Externas**	**Consultas Externas**
Baja	3	4	3
Media	4	5	4
Alta	6	7	6

Son 5 Entradas externas de complejidad baja cada una toma un valor de 4 el resultado de la multiplicación es el aporte que se obtiene.

Tabla 1.2 Complejidad por entradas.

	Complejidad			
	Baja	Media	Alta	Aporte
Entradas Externas	5			15
Salidas Externas		1		5
Consultas Externas	3	3		21
Archivos Lógicos Internos	8			56
Archivos de Interface Externos				
			Total	97

La suma total de los aportes da como resultado los puntos de función sin ajustar.

UFP= 15 +5+21+56 = 97

Cálculo del Factor de Ajuste y los Puntos de Función Ajustados.

Se basa en la cuantificación de ciertos coeficientes vinculados las características vinculadas del sistema, cada característica tiene una descripción y un peso o valor el cual se cuantifica.

Características a tener en cuenta.

Tabla 1.3 Coeficientes vinculados a las características del sistema.

Características	Descripción	Peso
Comunicación de datos	Aplicación Web	3
Procesamiento distribuidos de datos	Datos distribuidos	2
Rendimiento	No hay requerimientos especiales	0
Configuraciones utilizadas	No existen restricciones en el hardware.	0
Frecuencia de transacciones	Diaria	4
Entrada de datos On Line	Datos On Line	5
Eficiencia del usuario final	Media	3
Actualizaciones On Line	Las actualizaciones son On Line.	5
Procesamiento complejo	No hay lógico ni Matemático	0
Reusabilidad	Suple necesidades de los usuarios	4
Facilidad de instalación	Fácil, no existen restricciones	0
Facilidad de operación	Operación desatendida	5
Instalación en distintos lugares	Solamente en el servidor Web	2
Facilidad de cambio	Media	3

Con la suma de los valores (peso) se obtiene el grado total de influencia TDI.

TDI = 3 +2+4+5+3+5+4+5+2+3 = 36

El Factor de Ajuste se calcula AF mediante la siguiente ecuación.

AF = TDI x 0,01 + 0,65 = 36 x 0,01 + 0,65 = 1,01

Finalmente se obtienen los Puntos de Función Ajustados FP:

FP = UFP x AF = 102 x 1,01 = 103,02

Tabla 1.4 Factores de Escala.

FACTORES DE ESCALA		
Factores	**Valor**	**Justificación**
PREC	4.96	Precedencia baja
FLEX	3.04	Flexibilidad media
TEAM	1.10	Equipo cohesionado
RESL	2.83	Buena fortaleza de arquitectura
PMAT	7.80	Nivel 1 (Bajo)
ΣWi = 14,76		

Estimación de la cantidad de instrucciones fuente.

Tabla 1.5 Instrucciones Fuente.

INSTRUCCIONES FUENTES		
CARACTERÍSTICAS		**VALOR**
Puntos de Fusión Desajustados		97
Lenguaje	ASP	60%
	HTML	20%
	SQL	15%
	VBScript	5%
Instrucciones fuentes por puntos de fusión	ASP	65
	HTML	20
	SQL	30
	VBScript	40
Instrucciones fuentes		4753

Tabla 1.6 Factores Modificadores del Esfuerzo.

FACTORES MODIFICADORES DEL ESFUERZO		
Factores	**Valor**	**Justificación**
RCPX(RELY, DATA, CPLX, DOCU)	0,83	Base de datos simple y requiere de una gran complejidad debido al tratamiento de fichero en el lado del servidor y otros aspectos.
RUSE(RUSE)	0,96	No se reutilizan elementos.
PDIF(TIME,STOR,PVOL)	1,25	El tiempo y la memoria estimada para el proyecto son de alta complejidad.
PREX(AEXP,PEXP,LTEX)	1,58	Los especialistas no tienen

		experiencia en el uso de las tecnologías.
FCIL(TOOL, SITE)	0,99	Se han utilizado herramientas de alto nivel de desarrollo. Se ha usado notación UML.
SCED(SCED)	1	Los requerimientos de cumplimiento de cronograma son normales.
PERS(ACAP, PCAP, PCON)	1,06	Analista de alto nivel. El ingreso de personas anualmente se comporta de manera nominal y los programadores tienen una elevada capacidad.
$\prod_{i=1}^{n} EMi = 1,65142$		

Esfuerzo

PM = A x (size) $^{B\ x}$ PRODUCTO (i =1 ... n) EMi

Donde:

PM: Esfuerzo nominal requerido en meses hombres.

A = 2,94

C = 0,91

Size (tamaño): Miles Instrucciones Fuentes = 4,753

B = 0,91 + 0,01 * 14,76 = 1,0576

PM = 2,94 x $(4,753)^{1,0576}$ x 1,65142 = 25,24

PM ≈ <u>24 Hombres /Mes</u>

Cálculo de Tiempo de Desarrollo TDEV:

TDEV = D x PM^F

Siendo:

F = E + 0,2 X (B – C)

Donde:

D = 3,67

E = 0,28

C = 0,91

B = 1,0576 (exponente del tamaño en la fórmula de **PM**)

Entonces:

F = 0,28 + 0,2 * (1,0576-0,91) = 0,3095

TDEV = 3,67 x $25,24^{0,3095}$ = 9,84

TDEV ≈ 10 meses.

Cálculo de la cantidad de hombres CH:

CH = PM / TDEV

Por Tanto:

CH = 24 / 9,84 = 2,43 hombres

CH ≈ 2 hombres. El software fue elabora por 1 hombre

Costo:

El salario promedio es de $520

CHM = CH * Salario promedio

CHM = 1 * 520 = 520 $/meses

Entonces:

Costo = CHM * PM

Costo = 520 * 24 = $12 480

1.4.2 Análisis de beneficios tangibles e intangibles

Con la implantación del sistema automatizado SISCOIN para auxiliar el proceso del control y análisis de la información sobre los PFCM; se elevarán los niveles de eficiencia de este proceso, ganando en rapidez y calidad.

Los oficiales operativos que ingresaran en el MININT después de egresado de la SUM estarán mejores preparados para explotar las NTIC a lo largo del desempeño de su profesión.

El sistema está orientado al usuario, es de fácil aprendizaje, por lo que no reporta grandes gastos por concepto de entrenamiento de los trabajadores del negocio. No son necesarios los gastos por concepto de tecnología pues el MININT cuenta con los necesarios para la implantación de la aplicación incluida la conectividad con la red local.

Por todo ello se considera que es factible el desarrollo de la aplicación y que el "costo de desarrollo" está plenamente justificado.

1.5 Conclusiones

En el desarrollo del capítulo se abordo el entorno que se encuentra el sistema, la importancia de las TICs y su justificación en la solución del problema. Además se identifico el modelo del dominio propuesto y se llevo a cabo la estimación de costos de desarrollo del sistema con el uso de la herramienta COCOMO.

CAPÍTULO II. TENDENCIAS Y TECNOLOGÍAS ACTULES

2.1 Introducción

En el siguiente capitulo corresponde abordar los sistemas afines, así como la aplicación de la propuesta en el proceso educacional vinculado al estado del arte de la tecnología, haciendo énfasis en la justificación del empleo de las mismas ya que deben aportar el mayor beneficio posible durante su explotación en la solución de problemas.

2.2 Uso de las TICS en la educación.

Sentar las bases para la creación y el desarrollo de una cultura tecnológica en los actores del sistema de educación, que les permita comprender el papel de las tecnologías informática y Telemática en la generación de calidad, adoptar posturas críticas contra su uso irracional y utilizarlas racionalmente, en sus alcances apropiados.

Comprender y poder utilizar las nuevas tecnologías y asimilar los cambios que van a ocasionar en nuestro trabajo profesional y en nuestra vida en general, son parte de los cambios que habrán de producirse y conducirse inteligentemente en el mundo académico latinoamericano para producir calidad a través de esas tecnologías. Esta reflexión me conduce a pensar que será necesario un proceso de alfabetización informática y telemática de todos los actores que intervienen en los procesos de un sistema de educación, proceso que debe adaptarse a las características generacionales, profesionales y culturales de cada grupo de actores.

Este proceso de una nueva realidad educativa, implica entonces que estando en un contexto socio tecnológico, con estudiantes que vienen configurando una mente virtual, es necesario que también se efectúe una transformación de la educación centrada en el estudiante pero adecuando su nuevo entorno como un medio de aprendizaje natural.

Es evidente que estamos justo en un momento donde se avizora grandes cambios y transformaciones en la educación básica, justamente por estar inmersos en la Sociedad Red y por tener a la mano tecnologías modernas como son la computadora y el Internet que vienen configurando una serie de cambios, no sólo en aspectos contextuales sino principalmente en la forma de pensar, procesar información, realizar operaciones cognitivas, etc. en los estudiantes.

Estos cambios configuran todo un reto que el sistema educativo debe saber asumir con responsabilidad y direccionando adecuadamente el uso de las TIC por lo que se hace indispensable que la Educación Superior, se enfrasque en investigaciones pedagógicas capaces de utilizar estas tecnologías en el proceso docente educativo y en la preparación del claustro de profesores en los nuevos modelos pedagógicos que la educación cubana nos demanda.

En el caso del MININT que lleva un paso acelerado el uso de las TICS necesita personal calificado para poder llevar a cabo su correcta explotación, en el centro de instrucción recae el mayor peso o la mayor responsabilidad ya que ahí esta encavada la SUM MININT encargada de formar y superar al personal profesional.

Con el perfeccionamiento de la red de la institución mediante la fibra óptica elevando la velocidad de conexión y la puesta a punto de más de 300 clientes ligeros con 30 servidores con el objetivo que cada oficial tenga su puesto de trabajo para procesar la información, indica que los profesores y alumnos debemos llevar a cabo la tarea.

2.3 Portal de Enfrentamiento.

El portal de enfrentamiento se encuentra en el servidor de aplicaciones con el sistema operativo Windows 2003 server mediante al cual se accede por escritorio remoto, al identificarse con un usuario y contraseña carga la pantalla. Ver Anexo 5. Donde se encuentra los nombres con los vínculos a los distintos sistemas que lo integran. Para consultar cada uno de los sistemas debe autentificarse inicialmente.

Dentro de los software que lo integran están el de inmigración, búsqueda por direcciones, expediente del conductor entre otros. Los sistemas están hechos en diferentes plataformas y lenguajes de programación, unos en ambiente web y otros escritorio pero todos interactúan con el mismo gestor de base da datos ORACLE.

2.4 Herramientas, lenguajes y metodologías usadas para el desarrollo del sistema.

2.4.1 Tecnologías Web

El surgimiento de Internet ha traído como consecuencia que la comunidad internacional de programadores exija el surgimiento de herramientas y tecnologías que se integren a los nuevos requerimientos de la gran red de redes. Son importantes aspectos tales como velocidad de

procesamiento, integración con múltiples plataformas de trabajo, fácil desarrollo, acceso, mantenimiento, entre otros aspectos. A continuación se exponen algunas de las tecnologías consideradas **[4]**.

PHP

ASP

ASP.NET

JAVA SCRIPTS

2.4.2 Servidores Web

Un servidor Web es un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas Web o páginas HTML: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música **[3]**.

Algunos servidores Web importantes son:

APACHE.

Internet Information Server (IIS).

CHEROKEE.

Otros servidores, más simples pero más rápidos, son:

LIGHTTPD.

THTTPD.

2.4.3 Internet Information Server

IIS es la solución de Microsoft a las necesidades de la comunidad de programadores y usuarios, de enviar y recibir la información no sólo de sus clientes si no también en el entorno del negocio. Constituye la base esencial de un sistema de información destinado a la utilización de los servicios de Internet.

IIS engloba una serie de herramientas administrativas que permite controlar sitios Web, FTP, SMTP (correo saliente) y Servicio de noticias (news). Dispone también del soporte necesario para crear páginas dinámicas (ASP o ASP.NET), lenguaje de aplicaciones para Internet bastante extendido y que permite la conexión y acceso a bases de datos consiguiendo

aplicaciones Web dinámicas y escalables y dotando a nuestros sitios Web de un mayor rendimiento.

Contiene herramientas administrativas que puede utilizar para administrar redes, equipos, servicios y otros componentes del sistema. Simplifica la publicación de información en Internet o en la intranet, incluye una amplia gama de funciones administrativas para controlar sitios Web y el servidor Web. Con funciones de programación como páginas Active Server (ASP), puede crear e implementar aplicaciones Web flexibles y escalables.

2.4.4 ¿Por qué utilizar ISS?

La integración entre IIS, el sistema operativo Windows, ORACLE y el navegador Internet Explorer es óptima lo cual representa una elevada compatibilidad entre los diferentes componentes del proceso. Es recomendable usar las herramientas nativas de una plataforma, en este caso la que nos brinda Microsoft, con el fin de buscar un mayor rendimiento de las tecnologías usadas.

2.4.5 Características fundamentales de Dreamweaver

Dreamweaver posee varias características que lo hacen una herramienta muy potente para el desarrollo de sitios WWW. Ellas son:

- Se obtiene el control total sobre el código fuente, esto es gracias a la Split View (Vista dividida) que permite observar al mismo tiempo las vistas de código y de diseño.
- Se identifican fácilmente palabras claves y secuencias de comandos (scripts) en el código. El editor de texto integrado incluye coloreado del código ASP y JSP, sangrías automáticas y números de línea.
- Se maximiza la productividad con Server Behaviors (Comportamientos de servidor). Esta innovación, crea el formato y las secuencias de comandos del servidor, que se necesita para las aplicaciones de Web comunes, como actualizaciones e inserciones a las bases de datos.

Dreamweaver permite al usuario utilizar la mayoría de los navegadores Web instalados en su ordenador para pre visualizar las páginas Web. También dispone de herramientas de

administración de sitios dirigidas a principiantes como, por ejemplo, la habilidad de encontrar y reemplazar líneas de texto y código por cualquier tipo de parámetro especificado, hasta el sitio Web completo. El panel de comportamientos también permite crear JavaScript básico sin conocimientos de código.

Con la llegada de la versión MX, Macromedia incorporó herramientas de creación de contenido dinámico en Dreamweaver. En lo fundamental de las herramientas HTML WYSIWYG, también permite la conexión a Bases de Datos como MySQL y Microsoft Access, para filtrar y mostrar el contenido utilizando tecnología de script como, por ejemplo, ASP (Active Server Pages), ASP.NET, ColdFusion, JSP (JavaServer Pages), PHP sin necesidad de tener experiencia previa en programación.

2.4.6 Definición de ASP

Microsoft define ASP de la siguiente manera: "...es un entorno de secuencias de comandos en el lado del servidor que se puede utilizar para crear y ejecutar aplicaciones de servidor Web dinámicas, interactivas y de alto rendimiento...".

La filosofía de ASP resulta muy sencilla, en pocas palabras se puede definir de la siguiente forma: las páginas ASP son páginas que contienen código HTML, script de cliente y scripts que se ejecutan en el servidor, dando como resultado código HTML. Por lo tanto al cargar una página ASP en nuestro navegador, en realidad no estamos cargando la página ASP como tal, sino el resultado (respuesta) de la ejecución de la página ASP en forma de código HTML.

2.4.7 Aplicaciones ASP

Una aplicación basada en ASP consta de un directorio virtual en un servidor Web y de todos los subdirectorios y archivos contenidos en él. Al usar aplicaciones en ASP es posible mantener un estado, teniendo la capacidad de mantener información en dos tipos de estado:

- ✓ Estado de la aplicación, en la que toda la información relativa a una aplicación está disponible para todos los usuarios de la misma.
- ✓ Estado de sesión, en la que la información solo está disponible para un usuario o sesión específicos. Una sesión por lo tanto, pertenece a un solo usuario.

Para entender los aportes que ofrecen las páginas ASP se deben tener en cuenta una serie de características del protocolo HTTP (HyperText Transfer Protocol). Se dice que el protocolo HTTP es un protocolo sin estado, es decir, no se puede mantener un estado entre diferentes peticiones. Este se basa en el paradigma cliente - servidor o petición - respuesta. Se debe tener en cuenta lo siguiente:

- ✓ Después de realizar una petición, el cliente se desconecta del servidor y espera una respuesta. El servidor debe restablecer la conexión después de que haya procesado la petición.
- ✓ El servidor y el cliente sólo se tienen en cuenta durante la conexión. Por esta razón, ni el cliente ni el servidor pueden retener información entre diferentes peticiones o a través de diferentes páginas Web. Sin embargo, ASP permite al servidor almacenar información, o mantener el estado, entre las diferentes peticiones del cliente.

2.4.8 Objetos integrados en ASP

ASP contiene cinco objetos integrados que liberan al programador de la realización de tareas complejas: Application, Session, Request, Response y Server. Cada uno de ellos posee una serie de métodos y propiedades, no requieren que sean instanciados y además poseen una función determinada.

- ✓ **Request*:*** obtención de información del cliente.
- ✓ **Response*:*** envío de información al cliente.
- ✓ **Server*:*** control del entorno sobre el que se ejecuta ASP.
- ✓ **Session*:*** almacena información sobre la sesión de un usuario.
- ✓ **Application*:*** almacena información común para todos los usuarios de la aplicación ASP.

La sintaxis utilizada para poder acceder a los métodos y propiedades de los objetos depende del lenguaje de script que estemos utilizando, debido que el lenguaje de script que recomienda ASP es VBScript, no obstante, se puede utilizar el JavaScript.

2.4.9 ASP en comparación con tecnologías actuales

Ya hace un tiempo se ha venido observando un ascenso imparable en la utilización de PHP, puesto que cada día son muchísimas más las páginas web que se apoyan en este lenguaje para su funcionamiento.

PHP es un potente lenguaje de script del lado del servidor, que se utiliza principalmente para generar páginas de forma dinámica, atacando de una forma sencilla y nativa a diferentes bases de datos, por ser tan potente, se utiliza para muchísimas cosas más como: generar imágenes "on-the-fly", generar PDF, atacar diferentes protocolos de Internet (LDAP, IMAP, POP3, FTP, TELNET, JABBER,..), como lenguaje de script genérico, además, nos permite embeber sus pequeños fragmentos de código dentro de la página HTML y realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas programados íntegramente en un lenguaje distinto al HTML. Por otra parte, y es aquí donde reside su mayor interés con respecto a los lenguajes pensados para los CGI, PHP ofrece un sinfín de funciones para la explotación de bases de datos de una manera llana, sin complicaciones.

Como desventaja sobre el asp tenemos que tiene una peor integración con el paquete Office de Microsoft. Algunas de las diferencias principales pueden ser:

1. PHP, aunque multiplataforma, ha sido concebido inicialmente para entornos UNIX y es en este sistema operativo donde se pueden aprovechar mejor sus prestaciones. ASP, siendo una tecnología Microsoft, está orientado hacia sistemas operativos Windows, especialmente 2000.
2. Las tareas fundamentales que puede realizar directamente el lenguaje son definidas en PHP como funciones, mientras que ASP invoca más frecuentemente los objetos.

Para la realización de la aplicación, se escogió el ASP, debido a que la plataforma existente para el proyecto está montada sobre Microsoft (Windows 2000 Advanced Server) y por las prestaciones que nos brinda esta.

2.4.10 Ventajas de usar ASP

Hasta el momento hemos visto algunas comparaciones con otras tecnologías. A continuación se mencionan algunas de las ventajas que ofrece ASP:

1. ASP es totalmente gratuito para Microsoft Windows 2000, Windows *2000,* Windows *XP* o *Windows* 95/98.
2. Permite acceder a bases de datos de una forma sencilla y rápida.
3. Se ha portado a la plataforma Java por ChiliSoft y Halcyon Software, lo que permite que ASP sea usado en casi cualquier sistema operativo.
4. Las páginas se generan dinámicamente mediante el código de scripts, *(*guiones*)*.
5. El código de *script* se ejecuta en el servidor, y no depende del navegador que se emplee.
6. ASP permite usar componentes escritos en otros lenguajes (C++, Visual Basic, Delphi), que se pueden llamar desde los guiones ASP.
7. Desde una página ASP se pueden ejecutar servidores OLE en el servidor de web, lo que abre un abanico de nuevas posibilidades sólo accesibles previamente usando CGI y filtros ISAPI: acceso a bases de datos, acceso a ficheros, logging en el sistema, envío de correo, etc.

2.5 ¿Por que utilizar una Base de Datos?

- Es Compacto: No hacen falta archivos de papeles que pudieran ocupar mucho espacio.
- Es Rápido: La máquina puede obtener y modificar datos con mucha mayor velocidad que un ser humano. Así es posible satisfacer con rapidez consultas de casos particulares, del momento, sin necesidad de búsquedas visuales o manuales que refieren mucho tiempo.
- Es menos Laborioso: Se elimina gran parte del tedio de manera archivos a mano. Las tareas mecánicas siempre serán mejor realizadas por las máquinas.
- Es Actual: Se dispone en cualquier momento de información precisa y al día.
- Cómoda: Al tener la información en un mismo sitio, ahorraremos tiempo y trabajo.
- Disminuir la Redundancia: Definimos redundancia como la duplicación de datos.

Es fundamental hacer copia de seguridad de la base de datos cada vez que esta quede actualizada.

- Compartición de Datos: Hablamos de datos actuales, ya que al ser centralizados, se puede tener acceso a los datos con la última actualización en prácticamente tiempo real.

2.5.1 Tipos de Base de Datos

Los tipos de Base de Datos son tres:

- Base de Datos Jerárquica: Es aquella donde los datos se presentan en nivel múltiples que represente con raíz y sus ramificaciones.
- Bases de Datos Red: Es aquella que permite la conexión de los nodos en forma multidireccional, por lo que cada nodo puede tener varios dueños a la vez.
- Base de Datos Relacional: En informática, tipo de base de datos o sistema de administración de bases de datos, que almacena información en tablas (filas y columnas de datos) y realiza búsquedas utilizando los datos de columnas especificadas de una tabla para encontrar datos adicionales en otra tabla. En una base de datos relacional, las filas representan registros (conjuntos de datos acerca de elementos separados) y las columnas representan campos (atributos particulares de un registro). Al realizar las búsquedas, una base de datos relacional hace coincidir la información de un campo de una tabla con información en el campo correspondiente de otra tabla, y con ello produce una tercera tabla que combina los datos solicitados de ambas tablas.

2.5.2 Gestores de Base de Datos

Los Sistemas de gestión de base 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. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta **[1]**. Amplia es la lista de SGBD existentes entre los más reconocidos están:

- PostgreSQL.
- MySQL.
- Microsoft Access.

- Microsoft SQL Server.
- Oracle.

2.5.3 ¿Por qué utilizar Oracle 11G como Gestor de Bases de Datos?

Oracle es un Sistema de Gestión de Bases de Datos Relacional (SGBD-R) que implementa soluciones *eficientes y efectivas* para la manipulación de grandes volúmenes de información, que pueden estar disponibles para numerosos usuarios de forma simultánea. A partir de la versión ocho ***Oracle***, se ha introducido el modelo de Objetos como alternativa para almacenar la información **[2]**.

Utilizar Oracle permite obtener todas las ventajas que se derivan de trabajar con un modelo relacional, entre las más importantes: normalización, independencia, consistencia e integridad de los datos. Implementa SQL como lenguaje para el acceso a los datos, con PL/SQL como extensión procedural, que introduce la posibilidad de almacenar y ejecutar procedimientos en la base de datos (BD).

Como sistema de gestión está compuesto por un núcleo - Oracle Server – una BD y un conjunto de utilitarios y productos de software dirigidos a una amplia gama de roles usuario: administradores, diseñadores, programadores de aplicaciones y por supuesto a usuarios finales. Sus productos ofrecen una interfaz lo suficientemente amistosa para lograr buenos niveles de productividad y seguridad en la realización de las tareas, así como una aceptable estética en el caso de los medios para el desarrollo de aplicaciones.

2.5.4 Principales posibilidades que implementa Oracle Server

- Ambiente de trabajo cliente/servidor que posibilita el diseño y explotación de sistemas distribuidos de una forma segura y ordenada, lográndose el mismo nivel de transparencia y consistencia que en sistemas locales.
- Control total por parte de los diseñadores sobre la asignación de espacio para los diferentes objetos que conforman la BD.
- Optimizador de costo para el acceso a los datos basado en la recolección de estadísticas que reflejan el estado real de los datos.

- Procesamiento de transacciones que garantizan la consistencia y el mecanismo para lograr la concurrencia de numerosos usuarios, actuando tanto en lectura como en escritura sobre los mismos datos.
- Chequeo y control de las reglas de integridad e integridad referencial sobre todas las operaciones que modifican la BD.
- Amplias posibilidades para el control de acceso y auditoria, permitiendo la definición de roles y la asignación de manera selectiva.
- Mecanismos que brindan mucha seguridad y disponibilidad a los datos, permitiendo incluso la recuperación ante ciertos tipos de fallos, como son el mantenimiento de sistemas espejos en línea y la posibilidad de salvas calientes.

2.5.5 Elementos básicos de Oracle. Ficheros de Datos y Tablespaces.

Los ficheros de Datos en un sistema ORACLE constituyen el almacén físico de los datos (incluyendo el diccionario). Con un mínimo de un fichero (para el diccionario y las estructuras ORACLE) una BD puede contener múltiples ficheros. Los ficheros de datos no necesariamente tienen que ser contiguos, pero es muy beneficioso que el nivel de fragmentación que presenten en el disco sea bajo.

En la estructura física existen otros ficheros como son: ficheros de inicialización (init.ora), ficheros de control (Control File), ficheros de datos (Data File) y ficheros de reconstrucción (Redo Log)

Los ficheros de datos son asociados a estructuras lógicas llamadas Tablespace ver Figura 2.1. Un Tablespace se identifica por un nombre y puede contener uno o varios ficheros de datos y un fichero de datos a su vez puede estar asociado a un único Tablespace. El Tablespace es la entidad a través de la cual se administran los ficheros de datos. Todos los objetos de un sistema ORACLE residen en un Tablespace.

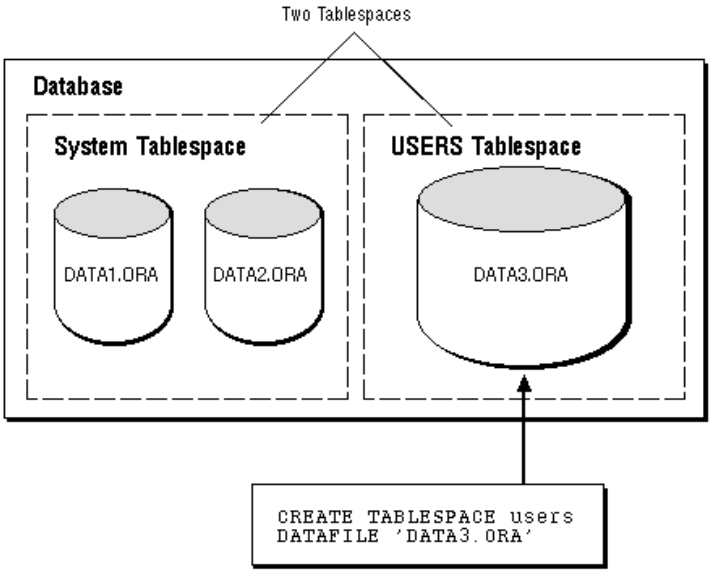


Figura 2.1 Ficheros de Datos Tablespaces.

Un Tablespace juega un rol importante para el buen desempeño de un sistema ORACLE: es la entidad que permite hacer el particionamiento de la BD así como controlar la ubicación física de los objetos que la conforman. Por tanto la cantidad, tamaño y ubicación de los mismos debe responder a una estrategia preestablecida de antemano por los diseñadores y el equipo de administración.

2.5.6 Objetos que conforman una Base de Datos Oracle

Una Base de datos Oracle está formada por un conjunto de *objetos* a través de los cuales interactúan los usuarios. Los usuarios crean y destruyen sus propios objetos para almacenar y manipular la información. Los objetos se identifican por un nombre y se agrupan en esquemas. Un esquema representa el conjunto de objetos de un usuario específico. Los usuarios tienen derechos plenos para manipular sus objetos, pero necesitan de permisos especiales para poder manipular objetos de otros usuarios.

Las tablas

Las tablas constituyen el objeto fundamental de una BD, en ellas se almacenan los datos. Las tablas están organizadas en filas y columnas. Cada fila representa una instancia de la entidad representada en la tabla y está formada por columnas que describen sus atributos. Una tabla

puede tener cualquier cantidad de filas. Los usuarios son responsables por insertar, actualizar y eliminar filas de las tablas.

Los Índices

Objetos asociados a columnas de una tabla que se utilizan como vía para llegar de forma rápida a una fila o conjunto de filas. Una tabla puede tener tantos índices como columnas tenga, y pueden existir índices que agrupen a más de una columna. Los índices se mantienen automáticamente, y aunque se asocian a una tabla son independientes de estas.

Las Vistas

Constituyen tablas virtuales, se crean a partir de una o varias tablas bases con el objetivo de imponer restricciones en cuanto a la información que se quiere mostrar o simplificar la forma de ver los datos

Los Sinónimos

Alias que se le puede asignar a un objeto. El nombre de cada objeto de la BD siempre está compuesto por el propietario del objeto, nombre del objeto. Por lo que el alias permite además de simplificar la referencia, ocultar el propietario.

Las Secuencias

Comúnmente en una BD se necesitan números aleatorios que no se repitan para individualizar entidades y reconstruir toda la información relacionada que se encuentra dispersa en distintas tablas. Para obtener estos números se utilizan las Secuencias. Pueden existir tantas secuencias como serie de números se necesiten. Las secuencias las utiliza el usuario según sus intereses.

2.5.7 Administración y seguridad

El DBA (administrador) de Oracle cumple responsabilidades para garantizar un adecuado funcionamiento del motor de base de datos y de las aplicaciones, dentro de las principales tareas del DBA se encuentran las siguientes:

- Garantizar la seguridad del sistema: el DBA debe siempre monitorear y administrar la seguridad del sistema. Esto involucra la incorporación y eliminación de usuarios, administración de espacios de discos (cuotas), auditorias y una revisión periódica para detectar probables problemas de seguridad.

- Monitorización: Debe monitorear continuamente el rendimiento del sistema y estar preparado para efectuar ajustes de sintonización de éste. En ciertas oportunidades esto involucra cambiar sólo algunos parámetros y otras veces reconstruir índices o reestructurar tablas.

- Respaldos: Consiste en desarrollar una estrategia afectiva de respaldos y recuperación de datos para mantener la estabilidad de toda la información guardada. La frecuencia de estos respaldos deberá decidirse dependiendo de la cantidad de procesos que alteran los datos a través del tiempo.

- Prevención de riesgos: Esta tarea se basa en calendarizar mantenciones a la base de datos (archivos lógicos) o cooperar en el mantenimiento de las máquinas al administrador del sistema. Debe eliminar problemas o situaciones potencialmente peligrosas.

2.5.8 Definición de Usuarios

ORACLE implementa un mecanismo de seguridad que permite realizar un control sobre los usuarios con un nivel de detalle o granularidad importante, regulando el acceso y utilización tanto de los objetos que conforman una Base de Datos como de los recursos compartidos del sistema a través de privilegios y límites.

Introduce dos conceptos que facilitan y flexibilizan la definición y mantenimiento de un esquema de seguridad: los roles y los profiles.

Un usuario se identifica con un nombre y una contraseña y tienen las siguientes propiedades:

Tablespace por defecto para la creación de sus objetos.

- ✓ Tablespace por defecto para sus temporales.
- ✓ Cuotas de capacidad de almacenamiento en los tablespaces.
- ✓ Privilegios que determinan sus posibilidades en el sistema.

- ✓ Un perfil de trabajo (profile) que regula el uso de recursos.

2.5.9 Privilegios

Un privilegio es un derecho que se le puede asignar a un usuario para ejecutar un tipo particular de instrucción SQL o para accesar algún objeto perteneciente al esquema de otro usuario.

Los privilegios se pueden otorgar de dos formas: explícitamente (GRANT privilegio TO usuario) ó indirectamente a través de roles (GRANT privilegio TO rol, GRANT rol TO usuario). Hay algunas acciones específicas que obligan a que un privilegio haya sido otorgado explícitamente.

Existen dos categorías de privilegios: privilegios de sistema y privilegios de objetos.

Privilegios de sistema: Los privilegios de sistemas agrupan a los privilegios que dan derecho para ejecutar acciones delicadas o que pudieran implicar riesgos, debido a su posible efecto destructivo o violatorio de la necesaria compartimentación de los datos, en un sistema o sobre algún objeto particular. Existen alrededor de 80, y se referencian por palabras reservadas. Su asignación debe ser muy cuidadosa y selectiva dada la potencia que le brindan al usuario que los recibe.

Algunos de los más significativos son CREATE TABLESPACE, DROP ANY TABLE (borrar cualquier tabla), etc.

Privilegios de objetos: Son derechos para ejecutar acciones sobre los objetos. Igualmente se referencian mediante palabras reservadas y existen en dependencia del tipo de objeto. Por ejemplo para los cluster's y trigger's no se definen privilegios. Para el caso particular de una tabla existen dos niveles de protección:

- Derechos para utilizar instrucciones DML (insert, update, delete, etc.)
- Derechos para utilizar instrucciones DDL (alter, index, references, etc.)

Otros objetos interesantes son los procedimientos en cualquiera de sus variantes (packages, stored, stand alone, etc.) y los trigger's. En ambos casos el dueño debe haber recibido

explícitamente los privilegios necesarios para ejecutar las acciones que se programan en el cuerpo. Todo el que con posterioridad los ejecute solo necesita el derecho de EXECUTE sobre el procedimiento.

2.5.10 Roles

Estructura lógica que permite agrupar privilegios (u otros roles) y que pueden ser otorgados a los usuarios. El concepto de rol permite organizar y facilitar la administración, dándole una gran flexibilidad. Con los roles se pueden categorizar a los usuarios de un sistema.

Cuando a un **rol** se le otorga o revoca algún privilegio, automáticamente esto toma efecto para todos los usuarios que tengan asignado este rol. A un usuario se le pueden otorgar múltiples roles.

Los roles permiten reunir a los usuarios en una sola unidad a la cual se le pueden aplicar permisos. Existen roles de servidor (servidor de aplicaciones) y de Base de Datos predefinidos, para tareas administrativas comunes, de manera que pueden asignársele determinados permisos administrativos a un usuario en particular. También se pueden crear roles de Base de Datos definidos por el usuario.
Roles fijos de la Base de Datos: Proveen agrupamientos con privilegios administrativos a nivel de Base de Datos.
Roles de usuarios definidos en la Base de Datos: También se pueden crear roles para Base de Datos, que representan un trabajo desarrollado por un grupo de empleados dentro de una organización, no es necesario asignar y quitar permisos a cada persona, en función de que cambia un rol, se pueden cambiar fácilmente los permisos del rol y hacer que los cambios se apliquen automáticamente a todos los miembros.

2.5.11 Reglas de integridad

Se conoce por reglas de integridad al conjunto de reglas definidas por los diseñadores de aplicaciones que deben cumplir los datos para que puedan ser considerados como válidos o correctos. Existe un numeroso conjunto de reglas preestablecidas que simplifican extremadamente la capacidad de garantizar su cumplimiento, por ejemplo NOT NULL para

garantizar que una columna tome algún valor, UNIQUE para impedir que en una tabla se inserten filas que dupliquen el valor de una columna, etc. En una aplicación pudieran existir reglas más complejas como serían las de integridad referencial y otras aún mucho más particulares que es necesario tener en cuenta igualmente durante el diseño.
ORACLE ofrece diversos métodos para implementar el chequeo de la integridad de los datos que van desde el control al nivel de las aplicaciones, los triger's de BD, hasta las restricciones de integridad al nivel del diccionario de datos o declarativas.

2.5.12 Restricciones de integridad Declarativas

Se conoce como restricciones de integridad declarativas (contrainst en la terminología ORACLE) al conjunto de reglas que un diseñador define para sus datos en el momento en que crea las tablas. Estas reglas se identifican por un nombre y se almacenan conjuntamente con la definición de las tablas **(**CREATE TABLE**)** en el diccionario de datos del sistema. Algunas reglas bajo ciertas circunstancias se pueden variar dinámicamente **(**ALTER TABLE**)**. Siempre que ORACLE detecte una violación en alguna restricción deshace la transacción **(**ROLLBACK**)** y emite el mensaje correspondiente a la sesión.

Este método declarativo de definir las restricciones de integridad tiene numerosas ventajas sobre cualquier otro, siendo las más significativas:

- Fácil de implementar, no se requiere de programación extra, constituyen cláusulas de un comando **SQL**.
- Fácil de administrar al estar centralizadas en un único lugar.
- Se logra una mayor productividad en la programación y mantenimiento de aplicaciones.
- Se alcanzan rendimientos superiores pues el optimizador de costo se auxilia de la semántica de estas reglas, como algo más que lo ayuda a escoger un camino óptimo para el acceso a los datos.

El chequeo de las restricciones de integridad por supuesto genera un overhead sobre el acceso a los datos, pero se considera que siempre será menor que en el que puede incurrir una aplicación que lo implemente de manera particular. Las restricciones declarativas se pueden

deshabilitar momentáneamente para ejecutar ciertas labores batch que generen sobre carga de movimientos (por ejemplo un import o sqlldr), esto le proporciona al DBA una gran flexibilidad en el desarrollo del trabajo.

2.5.13 Tipos de restricciones

ORACLE implementa un significativo conjunto de restricciones, prácticamente las más importantes, que brindan productividad para el diseñador. Por lo bien precisas que resultan vamos a describirlas individualmente.

NOT NULL

Fuerza a que una columna de una tabla tome algún valor, concretamente no admite nulos o ausencia de valores. Se pueden definir para todas las columnas que resulten necesarias.

UNIQUE KEY

Fuerza a no admitir filas que dupliquen el valor de alguna columna ya existente. Se pueden definir para todas las columnas de una tabla que resulte necesario o para una combinación de columnas concatenadas.

ORACLE para chequear esta restricción crea automáticamente un índice que nombra de igual forma que la restricción (cláusula CONSTRAINTS). El diseñador controla la asignación de espacio para este índice con las cláusulas STORAGE y TABLESPACE de la opción USING INDEX. En el ejemplo anterior se creará el índice hec_uniq en el Tablespace *datos* con un extent inicial de 20 MB.

PRIMARY KEY

Define una columna o combinación de columnas como llave primaria de una tabla. Una llave primaria no admite duplicados ni nulos, y se utiliza fundamentalmente para definir reglas de integridad referencial. En una tabla se pueden definir múltiples columnas como UNIQUE, pero sólo una como PRIMARY KEY. Para el chequeo de esta restricción ORACLE igualmente crea un índice con las mismas posibilidades que en UNIQUE.

2.5.14 Integridad Referencial

Por integridad referencial se denomina al conjunto de reglas que gobiernan las acciones que se toman al manipular las filas de tablas relacionadas, a través de columnas comunes en un esquema padre-hijo. Con manipulación estamos refiriéndonos a las operaciones de inserción, actualización y borrado.

FOREIGN KEY

Designa a la columna o combinación de columnas de una tabla como llave externa (foreign key) ,que debe ser referenciada por alguna llave primaria o única (PRIMARY KEY o UNIQUE que se denominará llave referenciada) en una relación padre-hijo. La tabla donde se define la llave externa pasa a ser la tabla hijo en la relación y la tabla que contiene la llave referenciada la tabla padre. Ambas tablas tienen que estar en el mismo servidor de datos y en la misma BD.

REFERENCES

Define la tabla padre de la relación y la columna que debe ser referenciada, por omisión se asumirá la llave primaria de la tabla.

ON DELETE CASCADE

Determina la acción que se tomará con las filas de una tabla hijo cuando se borre alguna fila referenciada en la tabla padre. Si se borra una fila del padre, todas las filas del hijo referenciadas se borrarán también de manera automática. Si esta opción no se especifica la acción anterior genera un error.

CHECK

La restricción **CHECK** permite definir explícitamente una condición. Útil para establecer relaciones más sofisticadas entre datos de una misma tabla. En la condición (con algunas limitaciones) se pueden utilizar los operadores y funciones empotrados en SQL y se puede referenciar cualquier columna de una tabla. Se pueden especificar múltiples restricciones para una misma columna.

2.6 Metodologías. Proceso Unificado de Desarrollo (RUP).

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process fue creado por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía con esta metodología era producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos.

Es un proceso dirigido por casos de uso, es decir que el proceso avanza a través de una serie de flujos de trabajo, como se muestra en la figura 2.2 que parten de los casos de uso; está centrado en la arquitectura y es iterativo e incremental.

Además cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software.

Divide en 4 fases el desarrollo del software:

- Inicio: El Objetivo en esta etapa es determinar la visión del proyecto.
- Elaboración: En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción: En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- Transmisión: El objetivo es llegar a obtener el reléase del proyecto.

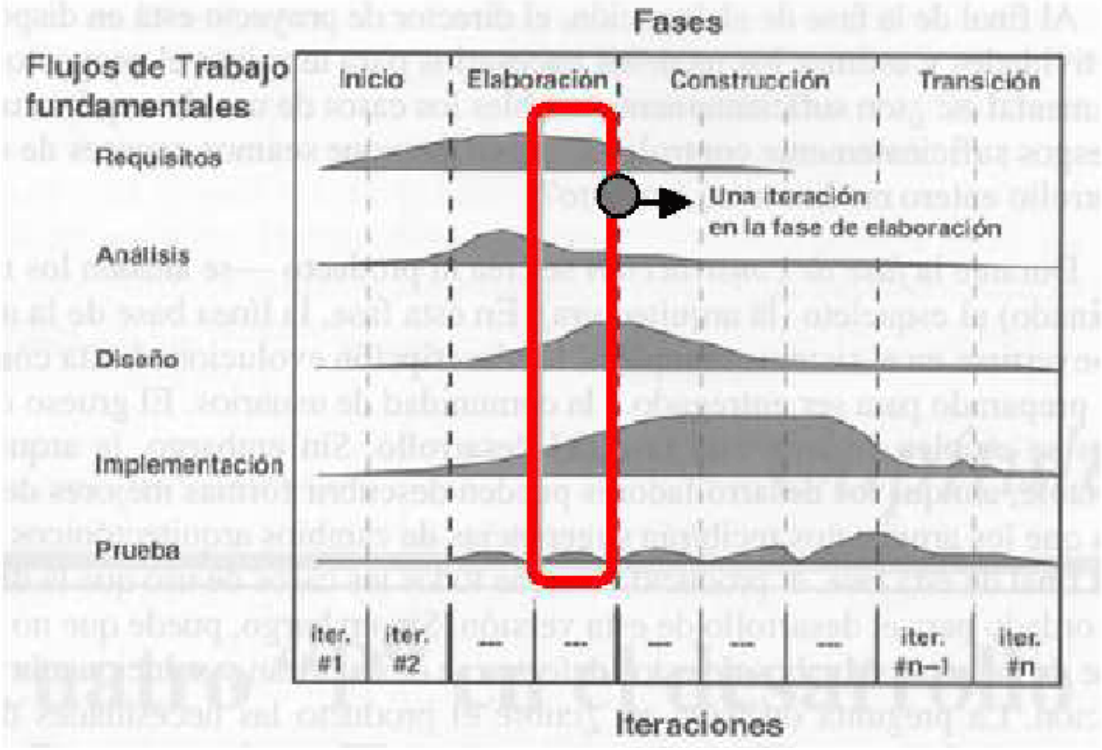


Figura 2.2 Flujos de trabajo de RUP (requisitos, análisis, diseño, implementación y prueba) tienen lugar sobre las cuatro fases.

2.6.1 Extreme Programing (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizada para proyectos de corto plazo. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Ver figura 2.3.

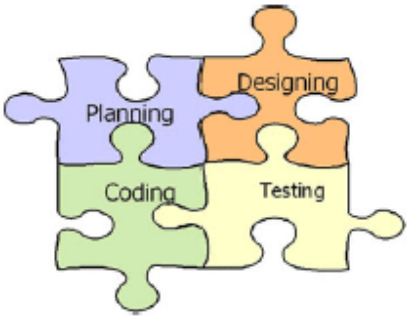


Figura 2.3 Metodología Extreme Programing (XP)

2.6.2 Microsoft Solution Framework (MSF)

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

Figura 2.4 Microsoft Solution Framework (MSF)

2.6.3 ¿Por qué se utilizó Rational Unified Process (RUP)?

Captura varias de las mejores prácticas en el desarrollo moderno de software en una forma que es aplicable para un amplio rango de proyectos y organizaciones. Es una guía de cómo utilizar de manera efectiva UML y le proporciona a cada miembro de un equipo un fácil acceso a una base de conocimiento con guías, plantillas y herramientas para todas las actividades críticas de desarrollo. Crea y mantiene modelos, en lugar de enfocarse en la producción de una gran cantidad de papeles de documentación.

2.7 Herramientas CASE

Las **Herramientas CASE** (Computer **A**ided **S**oftware **E**ngineering, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, calculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

Algunas aplicaciones case:

- MagicDraw.
- Visual Paradigm.
- Rational Rose Enterprise Edition.
- Enterprise Architect.

2.7.1 ¿Por qué se utilizó Rational Rose Enterprise Edition?

Es una de las mejores soluciones de modelado visual en el mundo, y está entre las mejores herramientas para traducir requisitos de alto nivel a una arquitectura flexible basada en componentes.

Rational se encuentra a la cabeza en cuanto al desarrollo del Unified Modeling Language (UML), que se ha convertido en la notación estandarizada empleada en Rational Rose para especificar, visualizar y construir desarrollos de software y sistemas.

Rational Rose domina el mercado de herramientas para el análisis, modelamiento, diseño y construcción orientada a objetos, tiene todas las características que los desarrolladores, analistas, y arquitectos exigen soporte UML incomparable, desarrollo basado en componentes con soporte para arquitecturas líderes en la industria y modelos de componentes, facilidad de uso e integración optimizada.

2.7.2 Lenguaje de Modelamiento Unificado (UML).

Para modelar el análisis y el diseño del software se utiliza el lenguaje UML que es uno de los más utilizado a nivel mundial en la actualidad.

UML (Unified Modeling Language, Lenguaje Unificado de Modelación). Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos del sistema de un Software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema. Un sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que finalmente beneficia a un usuario externo. El lenguaje de modelado pretende unificar la experiencia.

2.8 Conclusiones

En el capitulo se abordo la importancia de los servidores destinados a soportar la tecnología Web resaltando el Internet Information Server (IIS) utilizado en nuestra aplicación. Las ventajas de ASP y VBscripts como lenguaje de programación utilizado.

Se tomaron elementos sobre ORACLE como Sistema Gestor de Bases de Datos (SGBD) a utilizar, su modo de trabajo, arquitectura, modelo de seguridad, entre otros puntos a tener en cuenta a la hora de implementar una base de datos sobre ORACLE

CAPÍTULO III. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

3.1 Introducción

Este capítulo describe la construcción de la solución propuesta. Para ello muestra los Diagramas de Clases, el Diseño de la Base de Datos, el Modelo de Datos, el Diagrama de Despliegue e Implementación. A través de ellos serán definidos los componentes que formarán parte del sistema propuesto. Los Diagramas de Clase y de Componentes se muestran divididos por paquetes, logrando un mejor entendimiento por parte del lector sobre la solución propuesta.

Para la elaboración de los diagramas se utilizó la herramienta Rational Rose Enterprise Edition por sus bondades para la notación UML. Además se definen las principales pautas en el diseño de la interfaz para implementar el sistema propuesto.

3.2 Modelo de Diseño

En la fase de diseño se modela el sistema de manera que soporte todos los requisitos, tanto los funcionales como los no funcionales, creándose así una entrada apropiada para las actividades de implementación.

3.2.1 Patrones

"En la terminología de objetos, el patrón es una descripción de un problema y su solución que recibe un nombre y que puede emplearse en otros contextos; en teoría, indica la manera de utilizarlo en circunstancias diversas. Muchos patrones ofrecen orientación sobre cómo asignar las responsabilidades a los objetos ante determinada categoría de problemas. (…) Los patrones no se proponen descubrir ni expresar nuevos principios de la ingeniería de software. Todo lo contrario: intentan codificar el conocimiento, las expresiones y los principios ya existentes: cuanto más trillados y generalizados, tanto mejor."

3.2.2 Diseño de la Interfaz de la Aplicación

El éxito de un producto informático depende en buena medida del diseño gráfico de su interfaz **[8].**

El diseño Web no se limita a la apariencia estética, a la combinación de colores, o a un logo más o menos acertado. De él depende que la información sea útil, que los servicios se puedan

usar, es decir, el diseño convierte a una aplicación Web en algo atractivo para el usuario, por su estética y su utilidad.

Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos.

3.2.3 Estándares de la interfaz

Con vistas a lograr un diseño consistente de la interfaz de la aplicación, se ha seguido el esquema Cabecera-Navegador-Contenido, uno de los más usados actualmente. La interfaz para los usuarios de información y análisis son similares, la cabecera contiene el logotipo de la aplicación en la esquina superior izquierda (el primer lugar, según estadísticas, que mira el usuario al visitar una página Web) y una barra de navegación con enlaces de texto a las distintas secciones. En el área del contenido se muestra una imagen con los creadores del ALBA. La diferencia radica en que la interfaz del usuario de análisis solo tiene habilitado en el menú las opciones de recuperar.

La interfaz principal para la administración es diferente ya que en la misma el administrador crea, actualiza y borra los usuarios. Ver el Anexo 6. Interfaz de la Aplicación.

3.2.4 Requerimientos Funcionales

Los requerimientos funcionales son las capacidades que deben asegurar los sistemas para satisfacer al cliente o a los usuarios finales. Los requerimientos funcionales del sistema propuesto son:

R1. Autenticar el usuario en el sistema (abrir sesión).

R2. Crear usuarios.

R3. Modificar datos del usuario.

R4. Eliminar usuarios.

R5. Eliminar datos.

R6. Auditar la base de datos.

R7. Mostrar los usuarios que están conectados.

R8. Insertar datos (información) en el servidor de base de datos.

R9. Actualizar la información en el servidor de base de datos.

R10.Recuperar la información previamente almacenada según determinados criterios.

R11. Captar datos de autenticación.
R12. Buscar si existe el centro en la base.
R13. Registrar permisos de acceso de los usuarios.
R14. Habilitar opciones en el menú según usuario.
R15. Buscar información por diferentes criterios.
R16. Validar datos del usuario autenticado.
R17. Almacenar datos generales de los centros.
R18. Controlar la dirección IP de los usuarios que se conectan.

3.2.5 Requerimientos No Funcionales

Los Requerimientos No Funcionales son propiedades o cualidades que el producto debe tener, como restricciones del entorno o de implementación, rendimiento, entre otros aspectos.

Apariencia o interfaz externa:

La apariencia del sistema estará basada en los estándares definidos internacionalmente para una aplicación Web. La interfaz gráfica del sistema será legible y agradable, manteniendo un ambiente profesional. Será posible una fácil navegación por todos los módulos del sistema. La interfaz será funcional y un entorno interactivo que le permita al usuario el intercambio de información con el sistema.

Usabilidad: Es un producto que se puede adaptar fácilmente tanto a entornos educativos como empresariales Posee gran flexibilidad partiendo de que el conocimiento estructurado por temáticas puede ser organizado según las necesidades de la institución. Tendrá facilidades de uso para cualquier tipo de usuario, una interfaz consistente.

Rendimiento:

Rapidez en el procesamiento y en el tiempo de respuesta. Esta será garantizada por el servidor Web IIS y el SGBD ORACLE 11G.

Requerimiento de Soporte:

Garantizar la configuración del software y una instalación para asegurar los requerimientos de software del sistema. Se realizarán las pruebas de software para garantizar la calidad del producto.

Requerimientos de Seguridad:

Se ha definido niveles de usuarios para distribuir las responsabilidades del sistema. La información debe estar protegida de acceso no autorizado (confidencialidad).

Requerimientos de confiabilidad:

La información manejada por el sistema será objeto de cuidadosa protección contra estados inconsistentes de los datos *(*Integridad*)*. Se garantizará la protección brindando una herramienta segura. Además será posible la recuperación de información perdida mediante el sistema.

Requerimiento de ayuda y documentación en línea:

Mediante un menú inicial se podrá interactuar con el sistema según los privilegios del usuario. Será incluido un módulo de ayuda sobre el sistema.

Requerimientos del Software:

En los PC clientes Sistema Operativo Windows (Windows 2000, Advanced Server, XP, Server 2003, Linux) y algún navegador de Internet, preferentemente Internet Explorer 5+.

En el servidor Web Sistema Operativo Windows (Windows 2000 Server, Advanced Server, Server 2003) Internet Information Server.

En el servidor de base de datos Sistema Operativo Linux y SGBD ORACLE 11G

Requerimiento de Hardware:

Es necesario estar conectado a la RIM (Red Interna del MININT) mediante la implementación de los dispositivos de conexión como: Modem, Switch, radio enlace, fibra óptica y al menos un ordenador para la aplicación Web.

3.3 Modelo del sistema

En este epígrafe se muestra el modelo del sistema. Para ello se identifican los actores del sistema y se especifican los casos de uso del sistema.

3.3.1 Actores del sistema

Los actores de un sistema son agentes externos, es decir, aquellas personas o sistemas que interactúan con él. En la tabla 2.1 se definen los actores del sistema denominados con un nombre y una breve descripción de su responsabilidad y por qué interactúa con el sistema.

Tabla 3.1 Actores del sistema.

Actores	Justificación
Administrador	Es el encargado de administrar el sistema. Se comporta como un usuario pero por el rol que juega y los privilegios que tiene asignado lleva el control del resto de los usuarios y además le hace auditoria a la base de datos evitando la inconsistencia.
Oficial de información	Es el encargado de gestionar la información, mantener la base correctamente actualizada.
Oficial de análisis	Es el encargado de recuperar la información y procesar la misma.

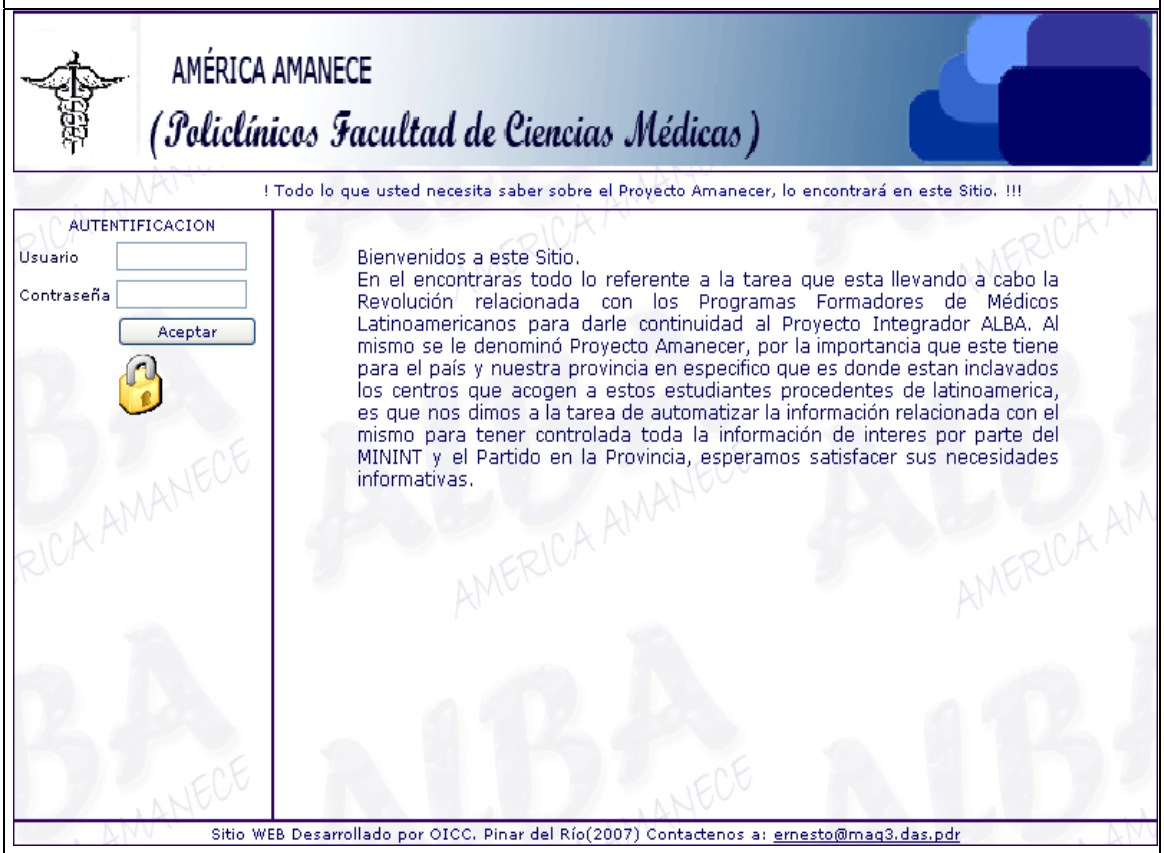
3.3.2 Modelo de casos de uso del Sistema

Los casos de uso son fragmentos de funcionalidad del sistema. En ellos se describe la secuencia determinada de eventos que realiza un actor en interacción con la aplicación, y se representan a través de los diagramas que evidencian gráficamente la relación entre los actores y los casos de uso. La descripción de los casos de uso del sistema puede consultarse en el Anexo 1 Modelo del Sistema.

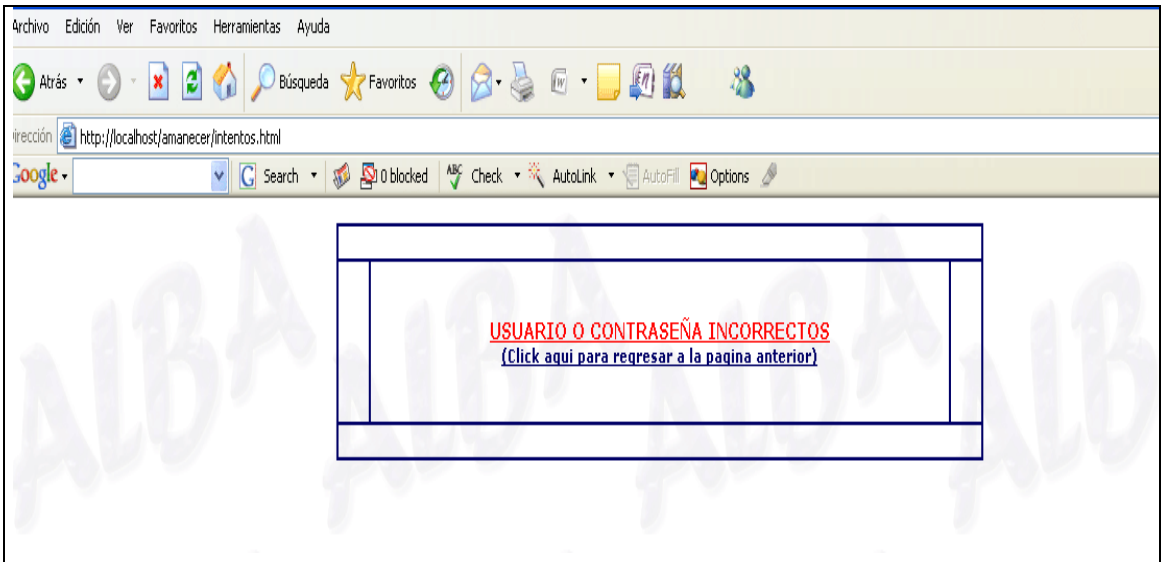
3.4 Descripción de los casos de uso del sistema.

3.4.1 Gestión de la Seguridad

Caso de uso:	Autentificar usuarios.
Actores:	Usuario (Inicia)
Propósito:	Gestionar los datos de los usuarios que acceden al sistema.
Resumen: El Caso de Uso se inicia cuando el usuario necesita autentificarse en el sistema. Valida los datos e inicia la sesión. El CU concluye cuando le asigna los permisos de entrada al usuario.	
Referencias:	R3,R13,R16
CU asociados:	
Precondiciones	El sistema se encuentra disponible y el (la) Usuario se autentifica.
Descripción	
Pantallas Asociadas	



Pantalla 1. Página de autentificación.



Pantalla 2. Página de Alerta.

Curso normal de los eventos	
Acción del actor	**Respuesta del sistema**
1 Se registra el usuario. Pantalla 1.	**2** Chequea los datos del usuario que se registro. **3** Si los datos son correctos redirecciona según el nivel del usuario autentificado. **4** Le muestra al usuario la pantalla principal de trabajo.
5 El usuario comienza a trabajar en el sistema.	**6** El caso de uso culmina con la autentificación del usuario.
Cursos Alternos	
Línea 2 **Sección Principal: Línea 2** Si el usuario no se registró correctamente (pantalla 1) no tendrá acceso al sistema, mostrándose la (pantalla 2) con una alerta brindando la posibilidad de retroceder a la página de autentificación.	
Sección Principal: Línea 10 Redireccionar los usuarios autentificados a sus respectivas pantalla de trabajo	

Sección Principal: Línea 11 No permitirles la entrada al sistema a los usuarios que no se registraron correctamente mostrándole la página de alerta.	
Post Condiciones	Autentificar los usuarios que acceden ala sistema.

Gestión de la información

Caso de uso:	Introducir Datos del Centro
Actores:	Oficial de Información (Inicia)
Propósito:	Gestionar la información de los Policlínicos Facultad de Ciencias Médicas (PFCM).
Resumen: El Caso de Uso lo inicia el oficial de información necesita y permite insertar los datos de los Policlínicos Facultad de Ciencias Médicas. El Caso de Uso finaliza una vez registrada la información.	
Referencias:	R8,R9,R12,R17
CU asociados:	Actualizar datos de los PFCM (include).
Precondiciones	El sistema se encuentra disponible y el (la) Oficial de Información se encuentra autentificado.
Descripción	
Pantallas Asociadas	

Pantalla 1. Página de Inicio del sistema.

Pantalla 2. Página de control de datos de los PFCM.

Pantalla 3. Página de actualización de los datos de los PFCM.

Curso normal de los eventos	
Acción del actor	**Respuesta del sistema**
1 El (la) Oficial de Información selecciona la opción datos del centro (elemento A) en el menú entrada de datos. Pantalla 1.	**3** Muestra la pantalla 2 para recoger los datos del PFCM a insertar por el Oficial de Información.
4 El Oficial de Información introduce todos los datos que se piden en la pantalla 3. **5** El Oficial de Información posteriormente pulsa el botón aceptar.	**6** Chequea que todos los datos obligatorios hayan sido introducidos por el oficial de información. **7** Verifica si los datos que se van a almacenar fueron introducidos con anterioridad. **8** Almacena los datos en la Base de Datos. **9** El CU culmina cuando la información quede registrada en la base de datos
Cursos Alternos	
Si el Oficial de Información pulsa el botón Cancelar de la Pantalla 2, el sistema retorna limpia el formulario (Borra los datos escritos).	
Sección Principal: Línea 56 Si al chequear los datos de carácter obligatorio no fueron introducidos en su totalidad, el sistema muestra la siguiente pantalla de error:	

Sección Principal: Línea 36

Si al chequear en la base de datos ya fueron introducidos los datos, muestra la siguiente pantalla de advertencia.

Si el Oficial de Información da click sobre el hipervínculo actualizar muestra la pantalla 3 de lo contrario retorna a la pantalla 1 y no hace nada.

Sección Principal: Entre Línea 36 y 69

En la pantalla 3 si el Oficial de Información cambia algunos de los datos que se muestran y presiona el botón actualizar, se actualizan los datos en la base de datos. Si presiona el botón cancelar se limpia el formulario y no hace nada.

Post Condiciones	Se gestiona la información en la base de datos.

Recuperar datos

Caso de uso:	Consultar información por centro
Actores:	Oficial de Análisis (Inicia)
Propósito:	Recuperar la información de cada centro previamente almacenada en la BD.
Resumen: El Caso de Uso lo inicia el Oficial de Análisis permitiéndole recuperar la información por centro. El CU finaliza cuando se muestra la información solicitada.	

Referencias:	R9,R10,R12,R15
CU asociados:	
Precondiciones	El sistema se encuentra disponible y el Oficial de Análisis se encuentra autenticado en el sistema.

Descripción

Pantallas Asociadas

Pantalla 1. Página de Inicio del sistema.

Pantalla 2. Página de selección del PFCM.

Pantalla 3. Datos generales del PFCM.

Pantalla 4. Incidencias del PFCM.

Pantalla 5. Hospitalizados del PFCM.

Pantalla 6. Plantilla del PFCM.

Pantalla 7. Nacionalidades por PFCM.

Pantalla 8. Problemas constructivos por PFCM.

Curso normal de los eventos	
Acción del actor	**Respuesta del sistema**
1 El (la) Oficial de Análisis selecciona la opción Por	**2** Muestra la pantalla 2 para seleccionar el PFCM del cual se quiere obtener la información.

centro (elemento B) en el menú Salida de datos. Pantalla 1.	
3 El (la) Oficial de Análisis despliega el combo box (pantalla 2) y selecciona el PFCM del cual desea conocer la información.	**4** Se muestra la información general del PFCM con un conjunto de botones que le permite ampliar la información. Pantalla 3.
5 El (la) Oficial de Análisis solicita la información especifica que necesite del centro presionando uno de los botones. Pantalla 3.	**6** Si elige el botón incidencia ir a la sección incidencias. **7** Si elige el botón hospitalizados ir a la sección hospitalizados. **8** Si el usuario presiona el botón plantilla ir a la sección plantilla. **9** Si el usuario elige el botón nacionalidad ir a la sección nacionalidad. **10** Si el usuario elige el botón problemas const. ir a la sección problemas constructivos.
SECCIONES	
Si el Oficial de análisis pulsa el botón cancelar de la pantalla 3,limpia el combo box (centro seleccionado).	
Sección Incidencias	
Sección Principal: Línea 127 El sistema elabora un reporte con los datos más relevantes de las incidencias ocurridas en el centro.	
Sección Hospitalizados	
Sección Principal: Línea 133 El sistema elabora un reporte con los datos de los estudiantes que están hospitalizados del centro así como el estado que se encuentran.	
Sección Plantilla	
Sección Principal: Línea 140	

El sistema elabora un reporte con la estadística de la plantilla que labora en el centro.	
Sección Nacionalidad	
Sección Principal: Línea 146 El sistema elabora un reporte estadístico con la cantidad de estudiantes por nacionalidades divididas en femeninas, masculinos y el total general.	
Sección Problemas Constructivos	
Sección Principal: Línea 151 El sistema muestra un reporte con cada uno de los problemas constructivos que no se han solucionado del centro.	
Post Condiciones	Muestra la información recuperada de la base de datos en pantalla.

3.5 Diagramas de Clases Web

La forma tradicional de modelar clases no es aplicable a la hora de diseñar una aplicación Web, es por esto que a finales de los 90, cuando el desarrollo de aplicaciones Web se hizo más importante, Jim Conallen hace uso de las facilidades de extensión brindadas por el UML para basado en este lenguaje modelar aplicaciones Web. Publica hacia junio del 99 varios artículos incluido "Modelling We*b* Applications Architectures with UML" donde describe la extensión de UML para el modelado de aplicaciones Web **[7]**.

Se elaboró un diagrama de clases Web para cada Caso de Uso del sistema de forma tal que facilite la comprensión de cómo se relacionan los distintos componentes en la realización de cada uno de ellos. Ver Anexo 2 Diagrama de clases web.

3.6 Clases Persistentes

El diagrama de clases persistentes es usado para modelar la estructura lógica de la base de datos, con clases representando tablas y atributos de clase representando columnas. Ver Anexo 3. Diagramas de Clases Persistentes.

3.7 Modelo de Datos

El diagrama del modelo de datos se corresponde con la representación física de la base de datos. Ver Anexo 4. Modelo de Datos.

3.8 Mapa de Navegación

Un mapa de navegación es la representación gráfica de la organización de la información de una estructura web. Expresa todas las relaciones de jerarquía y secuencia y permite elaborar escenarios de comportamiento de los usuarios. También grafica, de modo que todos los profesionales participantes en un proyecto lo tengan claro, diferencias entre páginas dinámicas, administrables o estáticas **[9]**.

Casi cualquier proyecto en formato web requiere graficar la información en un mapa de navegación, pero resulta especialmente necesario en sitios web que poseen un volumen importante de contenidos.

En consonancia con lo anterior, el usuario espera encontrar una representación gráfica que le permita comprender la estructura cognitiva e informativa del servicio de información. Teniendo esto como punto de partida, las posibilidades son enormes. Los diseñadores gráficos pueden encontrar aquí uno de los campos más provechosos, aunque se enfrentan a la tesitura de tener que incluir más texto del que desearían. Por otra parte, un mapa de navegación de un web no puede ser exhaustivo hasta la saciedad: no deberá incluir la representación gráfica de todos y cada uno de los nodos informativos existentes en el sistema. Se impone la definición de criterios de ponderación que establezcan ciertos límites. Otra cuestión a tener en cuenta es que tampoco puede ser un calco de la estructura física de los directorios reales existentes, sino la representación mental que se hace de la organización de la información, desde el punto de vista del usuario, al cual orienta.

La cuestión clave en la creación y desarrollo de un mapa de navegación es que el visitante acude a él precisamente porque no ha encontrado, en la página principal o tras una breve exploración del sitio, lo que está buscando. De ahí la necesidad de que el mapa sea lo suficientemente explicativo como para resolver tal problema, o al menos indicar claramente un posible trayecto hasta el objetivo que se persigue.

Se confeccionó un mapa de navegación por cada uno de los actores del sistema siendo más explicito con los usuarios que finalmente van a explotar el producto. Ver Anexo 7. Mapa de navegación

3.9 Modelo de Despliegue

El Modelo de Despliegue define la arquitectura física del sistema por medio de nodos interconectados. Estos nodos son elementos de hardware sobre los cuales se ejecutan los elementos de software. Se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. Ver Anexo 8. Diagrama de Despliegue.

La aplicación que se propone está basada sobre una arquitectura distribuida representada por tres nodos. El nodo “Aplicación Cliente” está representado por la computadora del usuario el cual contiene un navegador de Internet. Este recibe la información en lenguaje HTML enviado

desde el servidor y se encarga de comunicarse con el nodo que contiene la aplicación Web a través del protocolo HTTP. Mediante los controles mostrados en la pantalla el usuario interactúa con el sistema. En el nodo "Servidor Web" se atienden las solicitudes del cliente, se procesan y se les da respuesta. A su vez están contenidas las transacciones que se realizan sobre el servidor de base datos donde se encuentra la información, a través, del protocolo TCP/IP.

A continuación describimos las características de cada uno de los nodos que componen el modelo de despliegue y la red mediante la cual se comunican los nodos.

Una red de computadoras (también llamada red de ordenadores o red informática) es un conjunto de equipos (computadoras y/o dispositivos) conectados por medio de cables, señales, ondas o cualquier otro método de transporte de datos, que comparten información (archivos), recursos (CD-ROM, impresoras, etc.) y servicios (acceso a Internet, e-mail, Chat, juegos), etc.

En este caso se trata de una Red Cerrada o RIM (Red Interna del MININT), por su tamaño se clasifica en LAN (Local Area Network). Su extensión es amplia a lo largo de toda la provincia por los 14 municipios, por lo que de alguna manera podemos decir que esta conformada por varias LAN interconectadas formando una red provincial. La misma tiene en su composición varias topologías tanto físicas como lógicas en dependencia de las características del entorno donde se aplica. Destacar que nuestra provincia se encuentra conectada con el resto de las provincias del país.

En nuestro entorno utilizamos la tipología de red estrella, utilizando como dispositivos: fibra óptica y última milla.

La red en estrella, es la red de área local en la cual cada dispositivo, denominado nodo, está conectado a un ordenador o computadora central con una configuración (topología) en forma de estrella. Normalmente, es una red que se compone de un dispositivo central y un conjunto de terminales conectados. En una red en estrella, los mensajes pasan directamente desde un nodo al dispositivo central, el cual gestiona la redistribución de la información a los demás nodos.

El protocolo utilizado es el TCP/ IP con conexiones PPP y ATM en clase A.
Empleamos diferentes tipos de conexiones a través de diferentes medios de transporte.

- Enlaces punto a punto que están a una velocidad de 192kbps hasta 2mbps.
- Enlaces a 100Mbps a través de cables UTP.
- Enlaces a 1Gbps a través de Fibra Óptica
- Enlaces a través de radio enlaces a 2Mbps.

La velocidad de conexión entre los navegadores Web (PC Clientes) y el servidor Web (Internet Information Server) está en dependencia de los medios de transportes utilizados, en el caso del municipio Sandino donde se encuentran encabados los Policlínicos Facultades de Ciencias Médicas es de 1Mb. Con respecto a la SUM del MININT la velocidad de conexión 2 MB

La velocidad de conexión entre los servidores (aplicación, web (Internet Information Server) y base de datos Oracle 11G) es de 100Mbps. Respecto a los sistemas nacionales que los servidores se encuentran en la capital la velocidad de conexión es de 2Mbps.

3.9.1 Características principales de los servidores

- Servidor Web (Internet Information Server)

Microprocesador dual core a 3,4GHz
1Gb de memoria RAM
250Gb de disco duro

- Servidor de Aplicaciones

Microprocesador dual core a 3,4GHz
1Gb de memoria RAM
250Gb de disco duro

- Servidor de base de datos (Oracle 11G)

Dos microprocesadores dual core a 3GHz
3Gb de memoria RAM
Seis discos duro de 300Gb cada uno

3.10 Resultados de la consulta a especialistas para la comprobación del grado de validez del sistema que se proponen.

La consulta a los especialistas permitió arribar a un consenso respecto al nivel de importancia de los aspectos evaluados. La totalidad de los especialistas reconocen la relevancia del sistema SISCOIN en tanto a calidad y dinamización, señalando en el 100% de los casos que es muy adecuado, en tanto favorece a la economía de tiempo, de personal involucrado y de confiabilidad en el proceso de análisis de los datos obtenidos acerca de la información sobre los PFCM.

3.11 Conclusiones

En este capítulo se modeló la construcción de la solución propuesta, a través del modelo de diseño, de implementación y de despliegue, así como también el modelo de datos y se analizaron los patrones de diseño y los principios de diseño de la interfaz de la aplicación.

CONCLUSIONES

- El proceso de análisis de la información se puede realizar desde cualquier puesto de trabajo conectado a la RIM
- El sistema SISCOIN propuesto viabiliza el proceso de análisis de la información registrada sobre los PFCM.
- Según criterios de especialistas, el sistema permite elevar el nivel de eficiencia del proceso de análisis de la información registrada sobre los PFCM. Eleva la preparación de los estudiantes en la SUM MININT.

RECOMENDACIONES

- Generalizar el sistema propuesto a todas las provincias donde existan estos tipos de centros escolares (PFCM).
- Mantener actualizada la base de datos con la información de los PFCM.
- Profundizar en el perfeccionamiento del sistema propuesto a partir de búsquedas temáticas por determinados criterios sobre la información registrada.
- Explotar todos los sistemas que componen el portal de enfrentamiento en la SUM MININT.

REFERENCIAS BIBLIOGRÁFICAS

[1] Korth H.F y Silberschatz A. Database Systems Concepts. McGraw-Hill, 1986. Traducción castellano: Fundamentos de Bases de Datos. McGraw-Hill, 1987.

[2] Wikipedia. La Enciclopedia Libre. Oracle. URL:http://es.wikipedia.org/wiki/Oracle

[3] Vegas, J. "Introducción a las aplicaciones Web". URL: http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node11.html.

[4] Álvarez, Miguel Ángel. DesarrolloWeb.com. Mayo de 2004. URL: http://www.desarrolloweb.com/articulos/

[5] Jacobson, I.; Booch, G.; Rumbaugh, J. "El Proceso Unificado de Desarrollo de Software". 2 Tomos. Addison Wesley. 2000.

[6] COCOMO II Model Manual. http://sunset.usc.edu/research/COCOMOII/ cocomo_main.html

[7] Conallen, Jim. "Building Web Applications with UML". Addison Wesley. 1999.

[8] Corsi, M. "Ajax: El Nuevo Abordaje De Interfaz De Desarrollo De Diseño Web Del que Todos Hablan". URL: http://www.masternewmedia.org/cgi-bin/mttb.r752.cgi/3103.

[9]Object Management Group, 2001. "OMG Unified Modeling Language Specification". URL: www.omg.org

BIBLIOGRAFÍA

Álvarez, Miguel Ángel. DesarrolloWeb.com. Mayo de 2004.
URL: http://www.desarrolloweb.com/articulos/

Cecconi, Pablo. Aplicaciones Web Espaciales con Software Libre. URL:
http://mapa.buenosaires.gov.ar/sig/AplicacionesWebEspacialesConSoftLibre.html

Connalen, Jim. "Building Web Applications with UML". Addison Wesley. 1999.

Corporación Autónoma Regional de Santander. URL: http://sig.cas.gov.co/

Corsi, M. "Ajax: El Nuevo Abordaje De Interfaz De Desarrollo De Diseño Web Del Que Todos Hablan".
URL: http://www.masternewmedia.org/cgi-bin/mttb.r752.cgi/3103.

Jacobson, I.; Booch, G.; Rumbaugh, J. "El Proceso Unificado de Desarrollo de Software. 2 Tomos. Addison Wesley. 2000.

Korth H.F. y Silberschatz A. Database Systems Concepts. McGraw-Hill, 1986.
Traducción castellano: Fundamentos de Bases de Datos. McGraw-Hill, 1987.

Mariaca, C. Introducción a las Aplicaciones Web.
URL:http://glud.udistrital.edu.co/glud/areas/doc/articulos/1_articulo_ws/.

Martínez Prieto, Ana Belén**.** "Introducción a los SGBDOO". Universidad de Oviedo. Mayo de 2004. URL: http://di002.edv.uniovi.es/~belen/jortoo96.html

Musciano, C y Hill K. "HTML, La guía completa". 2da Edición. McGraw-Hill, 1999
.

Notario de la Torre, Ángel: "Apuntes para un Compendio sobre Metodología de la

Investigación Científica". Cuba. 1999.

Letelier Torres, Patricio "Desarrollo de Software Orientado a Objeto usando UML". Departamento Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, España. Mayo, 2004. URL: http://www.creangel.com/uml/intro.php

Object Management Group, 2001. "OMG Unified Modeling Language Specification". URL: www.omg.org

Vegas, J. "Introducción a las aplicaciones Web". URL:http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node11.html.

Wikipedia. La Enciclopedia Libre. Oracle. URL: http://es.wikipedia.org/wiki/Oracle

Wikipedia. La Enciclopedia Libre. SGBD. URL: http://es.wikipedia.org/wiki/SGBD

ANEXOS

Anexo 1. Modelo del Sistema.

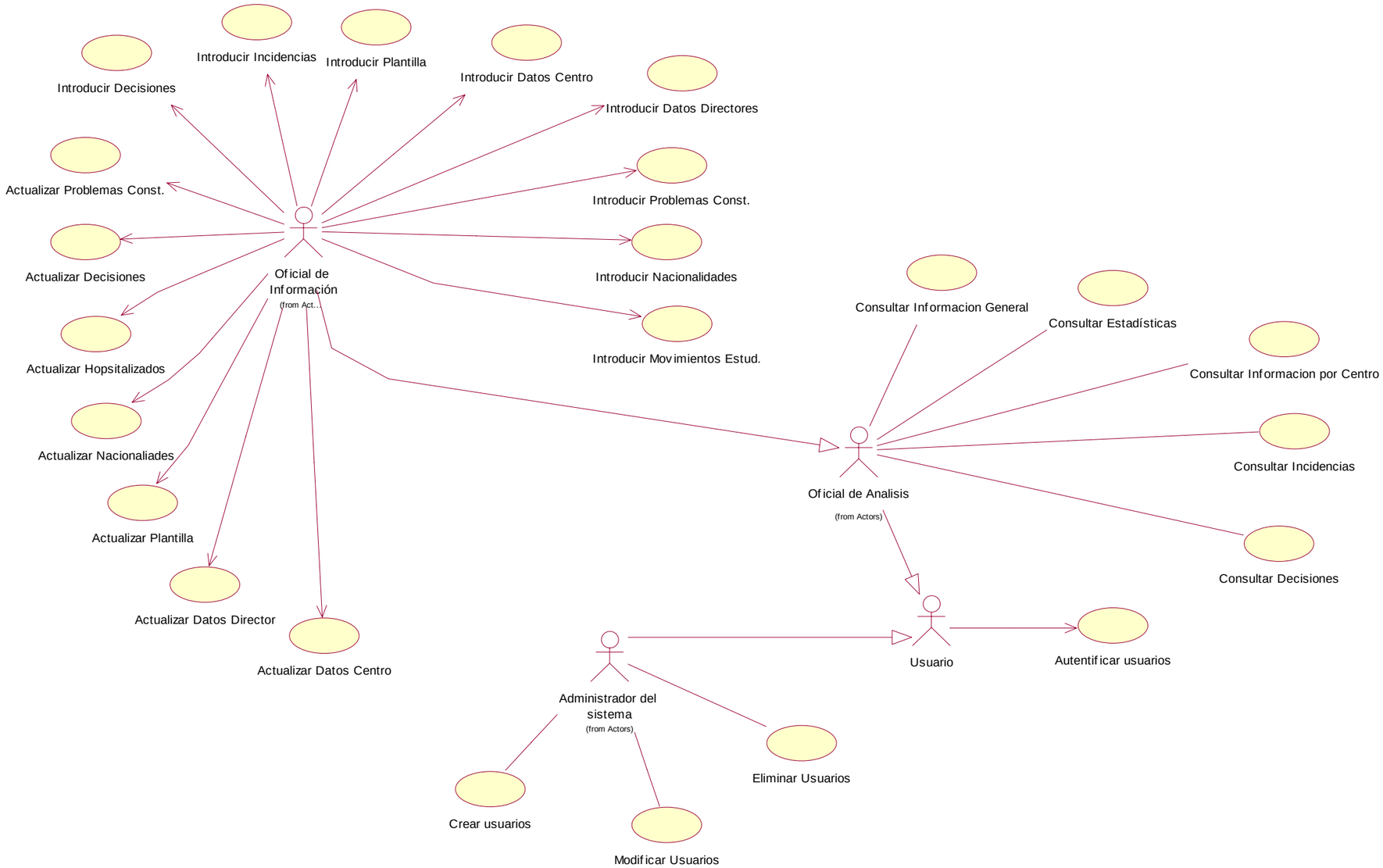


Figura 2.4 Modelo del Sistema

Anexo 2. Diagrama de Clases Web

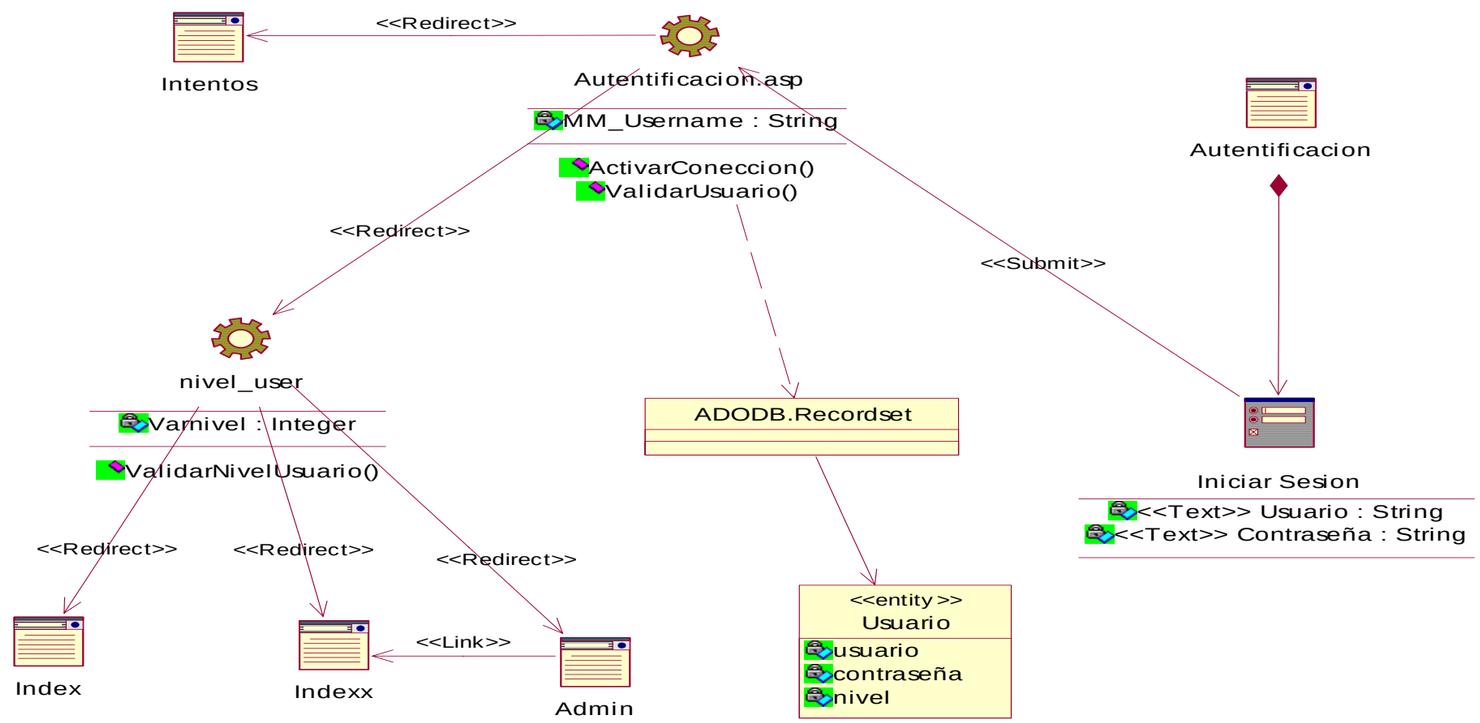


Figura 3.1 Diagrama de Clases Web. Caso de Uso Autentificar Usuarios.

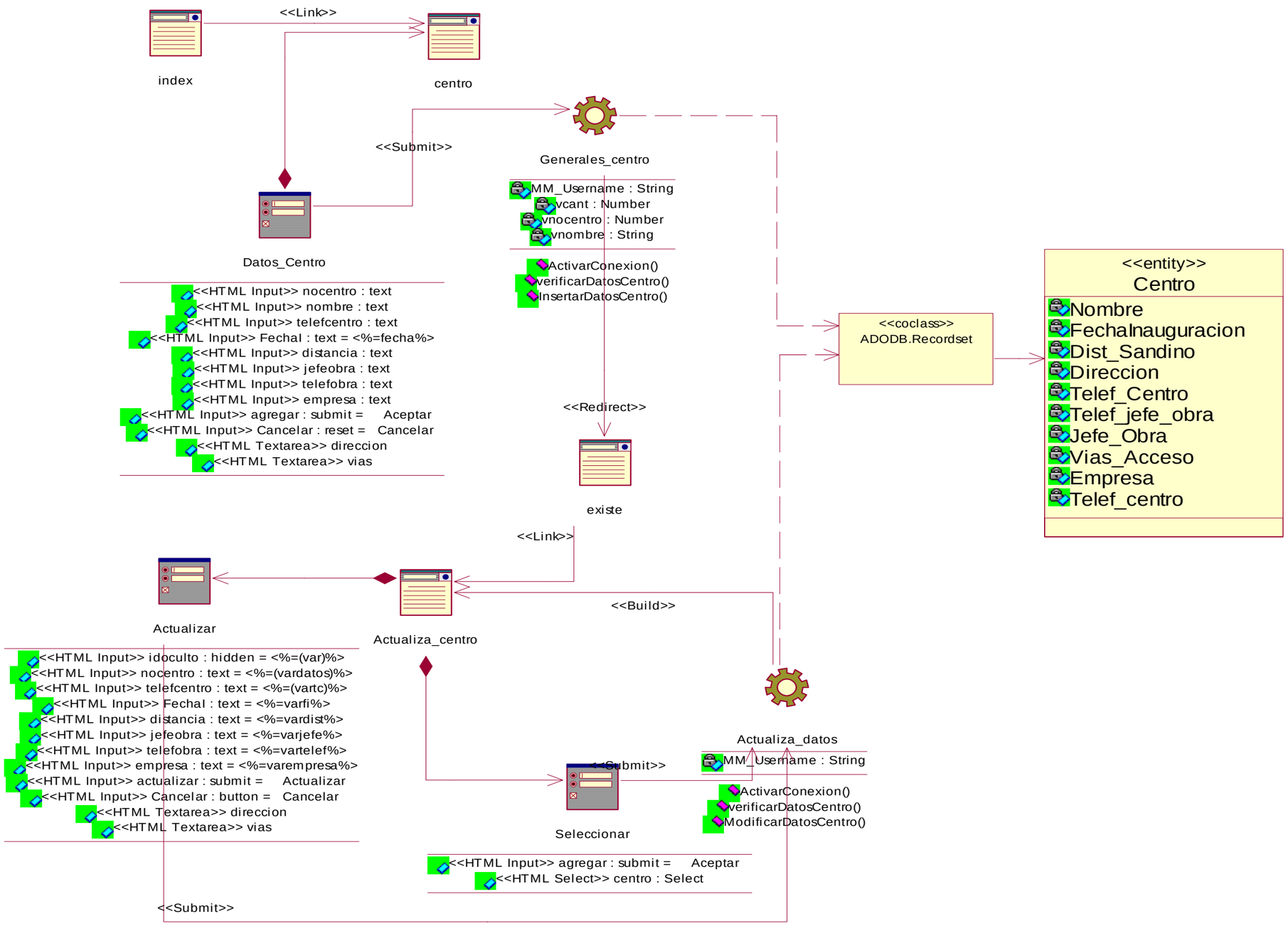


Figura 3.2 Diagrama de Clases Web. Caso de Uso Introducir Datos del Centro.

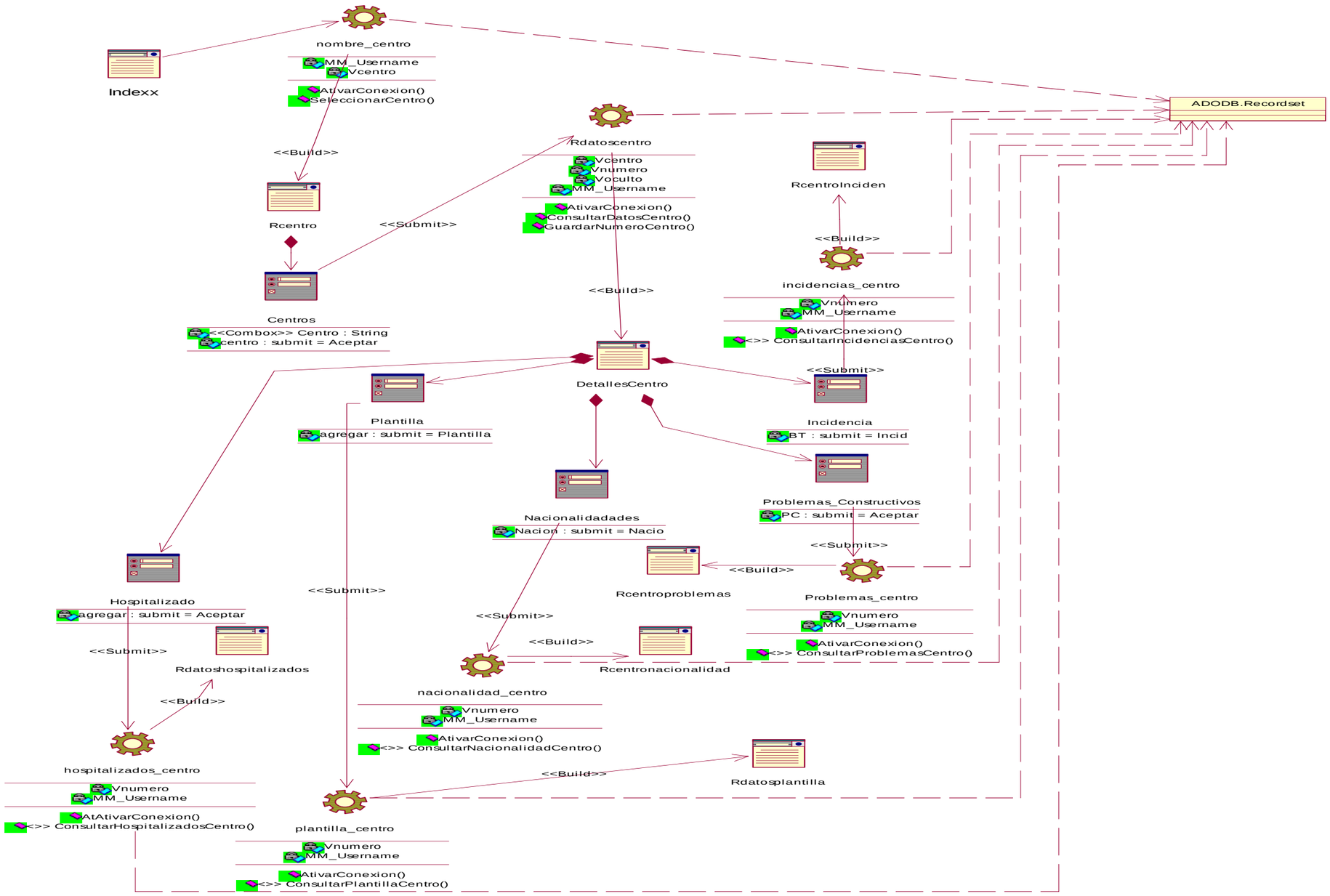


Figura 3.3 Diagrama de Clases Web. Caso de Uso Consultar Información por Centro.

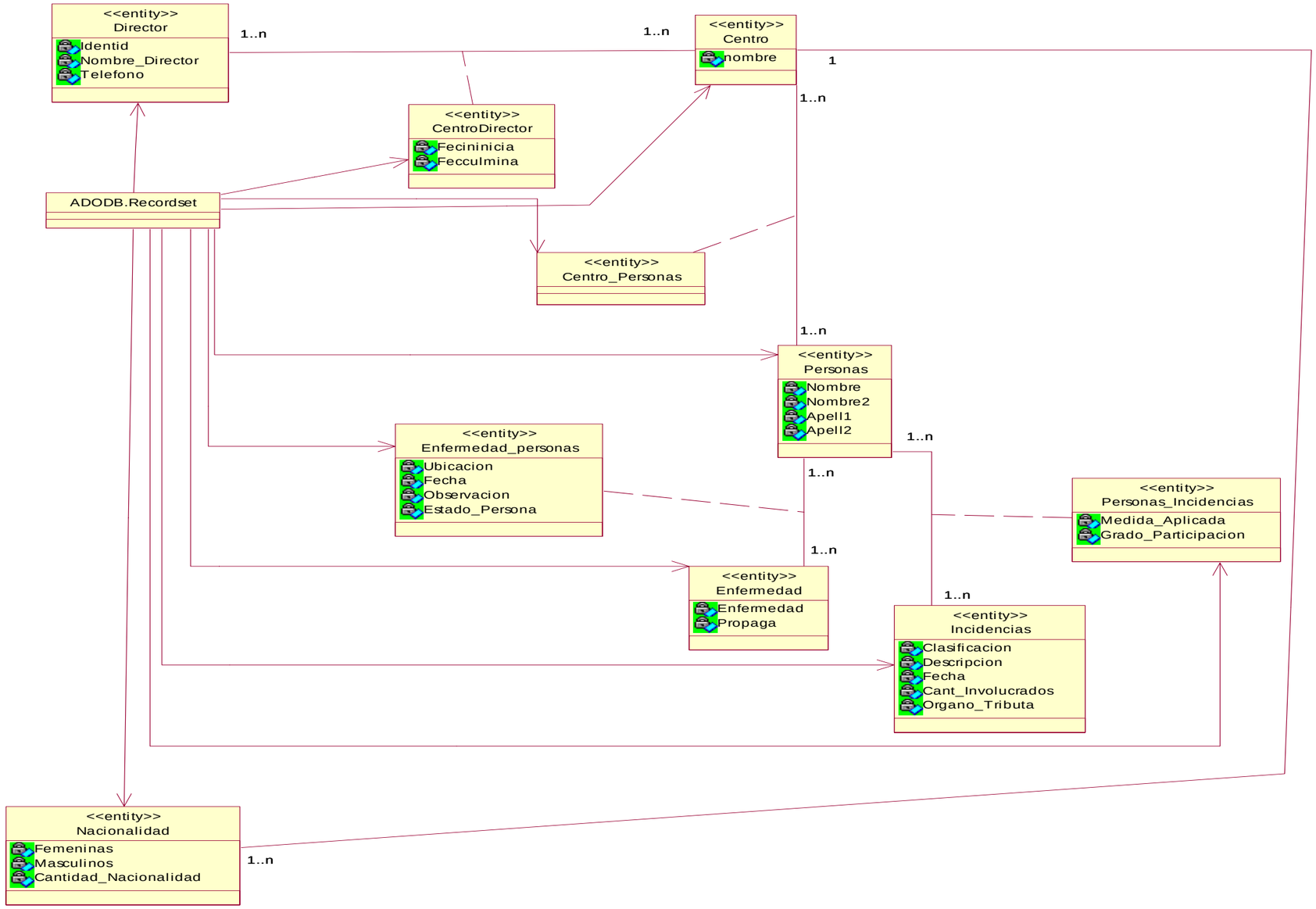


Figura 3. 4 Continuación del Diagrama de Clases Web. Caso de Uso Consultar Información por Centro.

Anexo 3. Diagrama de clases Persistentes.

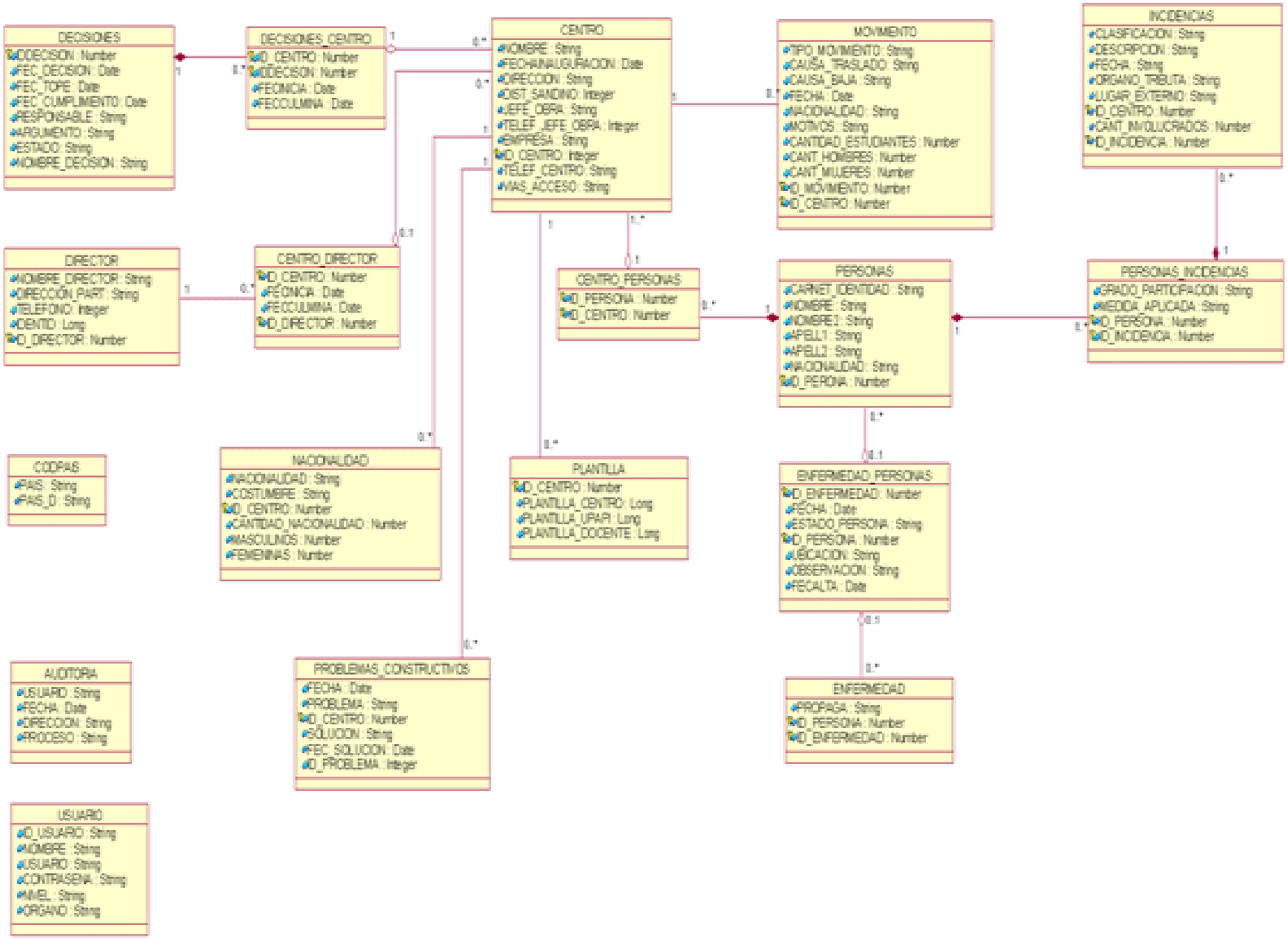


Figura 3.5 Diagrama de clases Persistentes.

Anexo 4. Modelo de Datos.

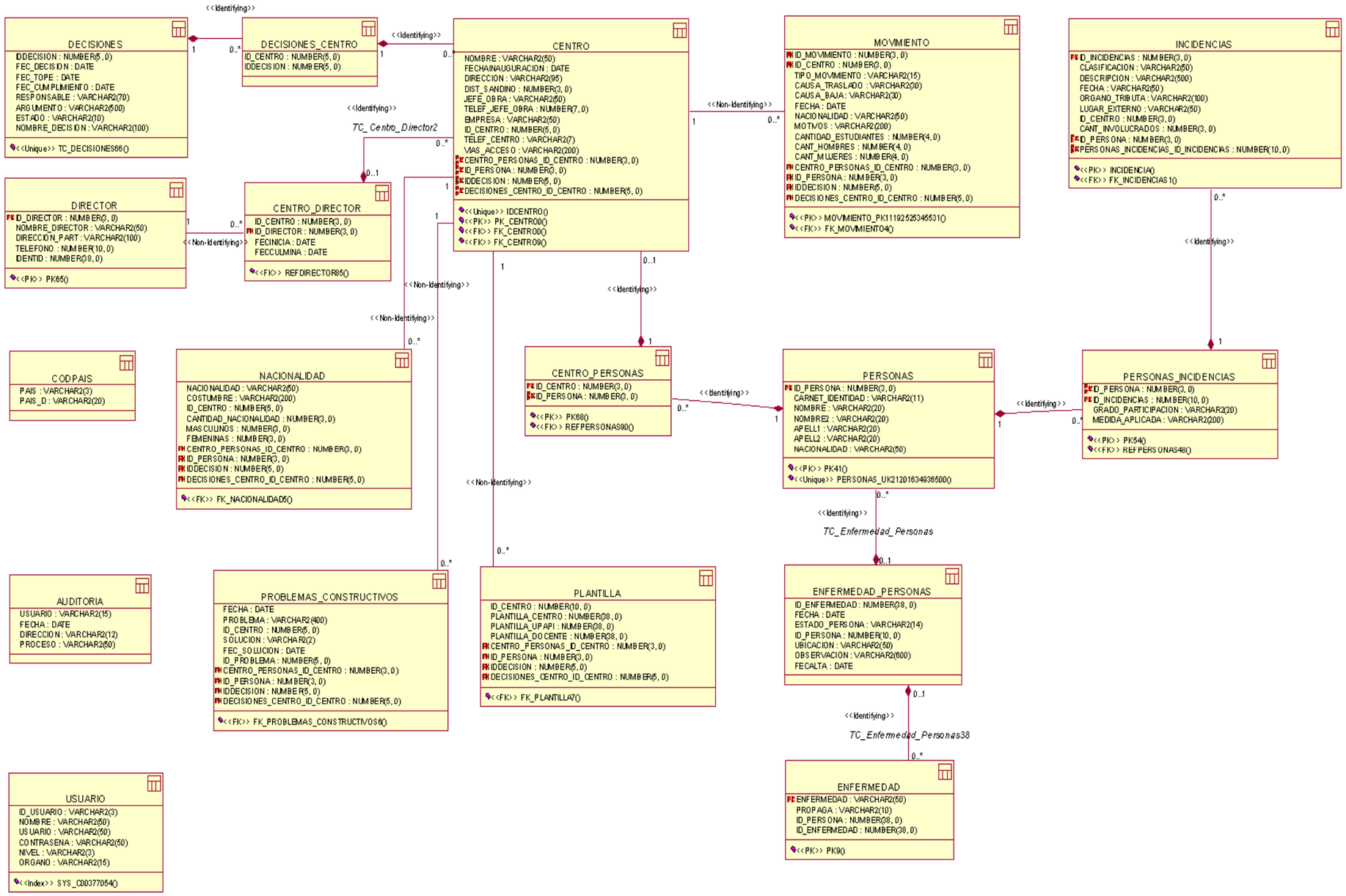


Figura 3.6 Modelo de Datos.

Anexo 5 Mapa de Navegación

Anexo 6. Interfaces principales.

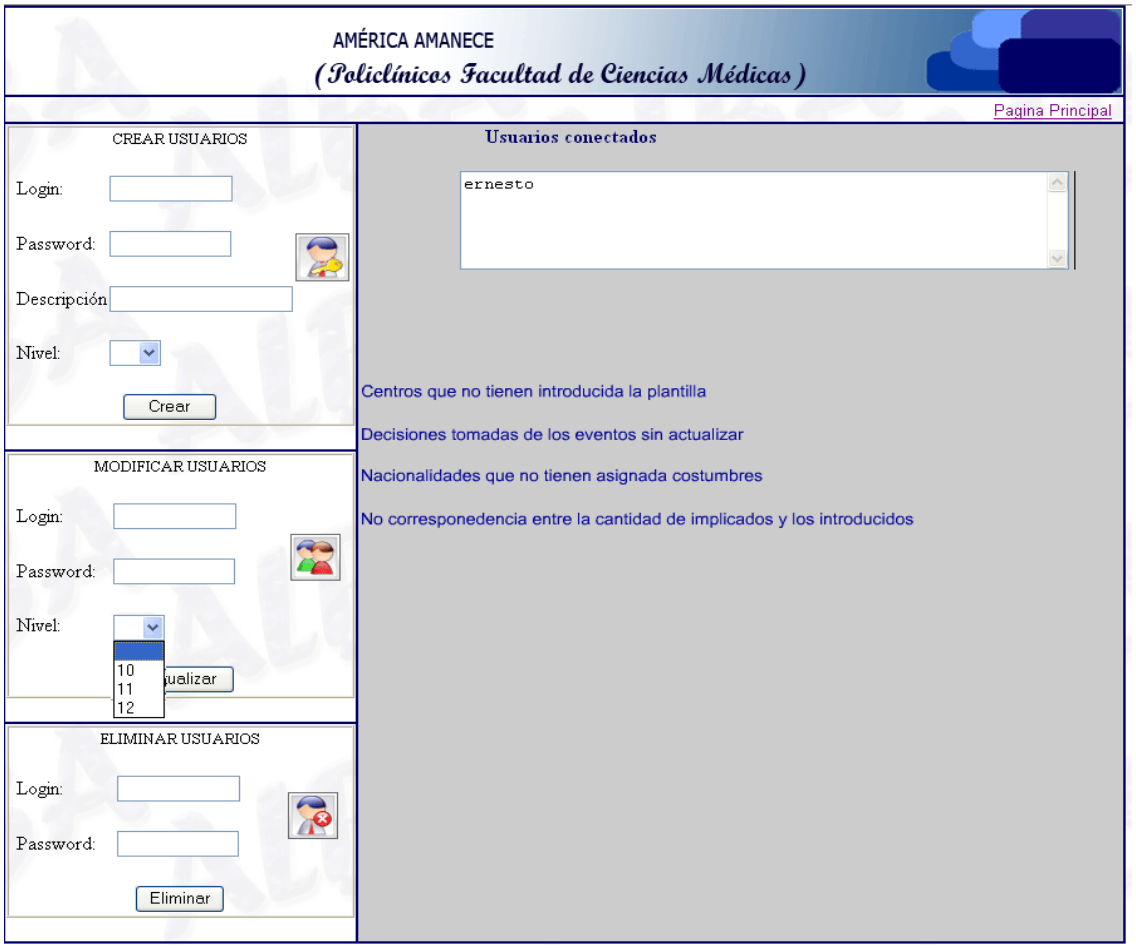


Figura 3.7 Interfaz principal para el Administrador.

Interfaz principal para la administración, el administrador tiene asignado nivel 10 (alta prioridad). Roles para crear, modificar y eliminar usuarios; los oficiales de información tienen nivel 11 (media prioridad) para acceder a todas las opciones del menú y los oficiales de análisis tienen nivel 12 (baja prioridad) para recuperar información.

Además mediante esta interfaz el administrador puede auditar la base de datos permitiéndole alertar a los usuarios sobre errores o inconsistencia en la BD.

Figura 3.8 Interfaz principal para los Oficiales de Información.

Es la página o interfaz que se les muestra a los Oficiales de Información después de autentificarse. El acceso a las distintas opciones del menú es limitado por lo que no todos los usuarios que se conecten tienen permiso para insertar o actualizar datos. La interfaz del administrador es diferente ya que brinda la posibilidad de crear, actualizar y borrar usuarios además de auditar la base de datos.

A Ayuda del sistema, tiene un hipervínculo a una página HTML la cual les describe a los usuarios detalladamente como trabajar en el sistema.

B Entrada de datos, menú en el que se encuentran las distintas opciones para poblar las tablas.

C Actualización de datos, el mismo contiene las opciones con hipervínculo a otras interfaces que permiten actualizar los datos previamente introducidos.

D Salida de datos, permite realizar las consultas (mostrar la información) de interés para los usuarios mediante páginas servidoras.

Figura 3.9 Interfaz principal para los Oficiales de Análisis.

Los Oficiales de Análisis solamente tienen acceso a recuperar la información previamente almacenada por lo que solamente tienen habilitado la Ayuda A y a los elementos del submenú Salida de datos D El resto de opciones del menú aparecen deshabilitadas.

Anexo 7 Mapa de Navegación

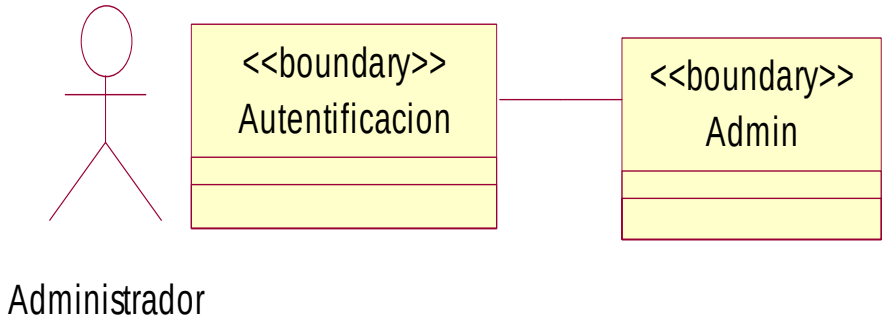


Figura 3.11 Mapa de navegación del usuario administrador

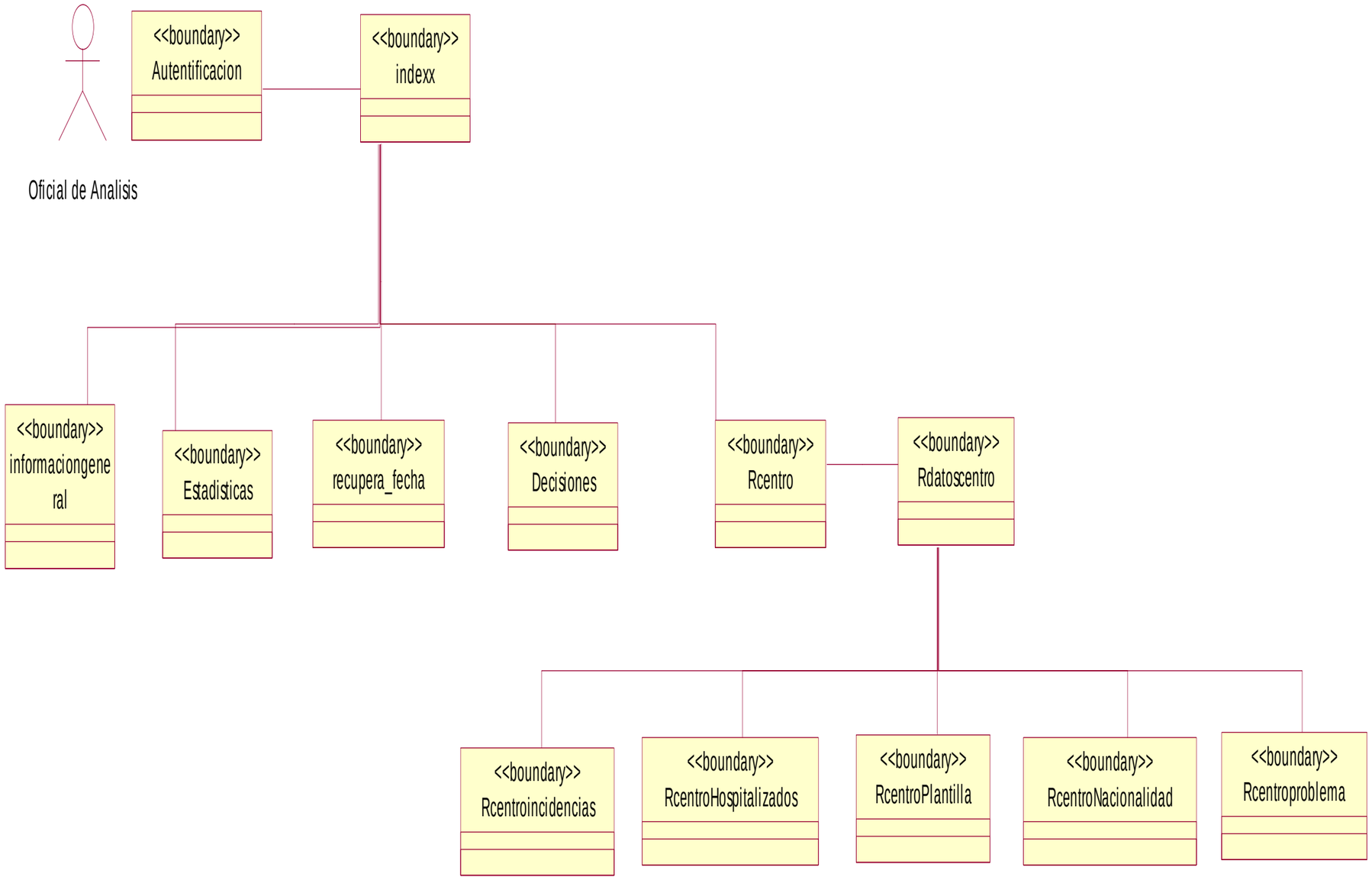


Figura 3.12 Mapa de navegación del usuario de análisis.

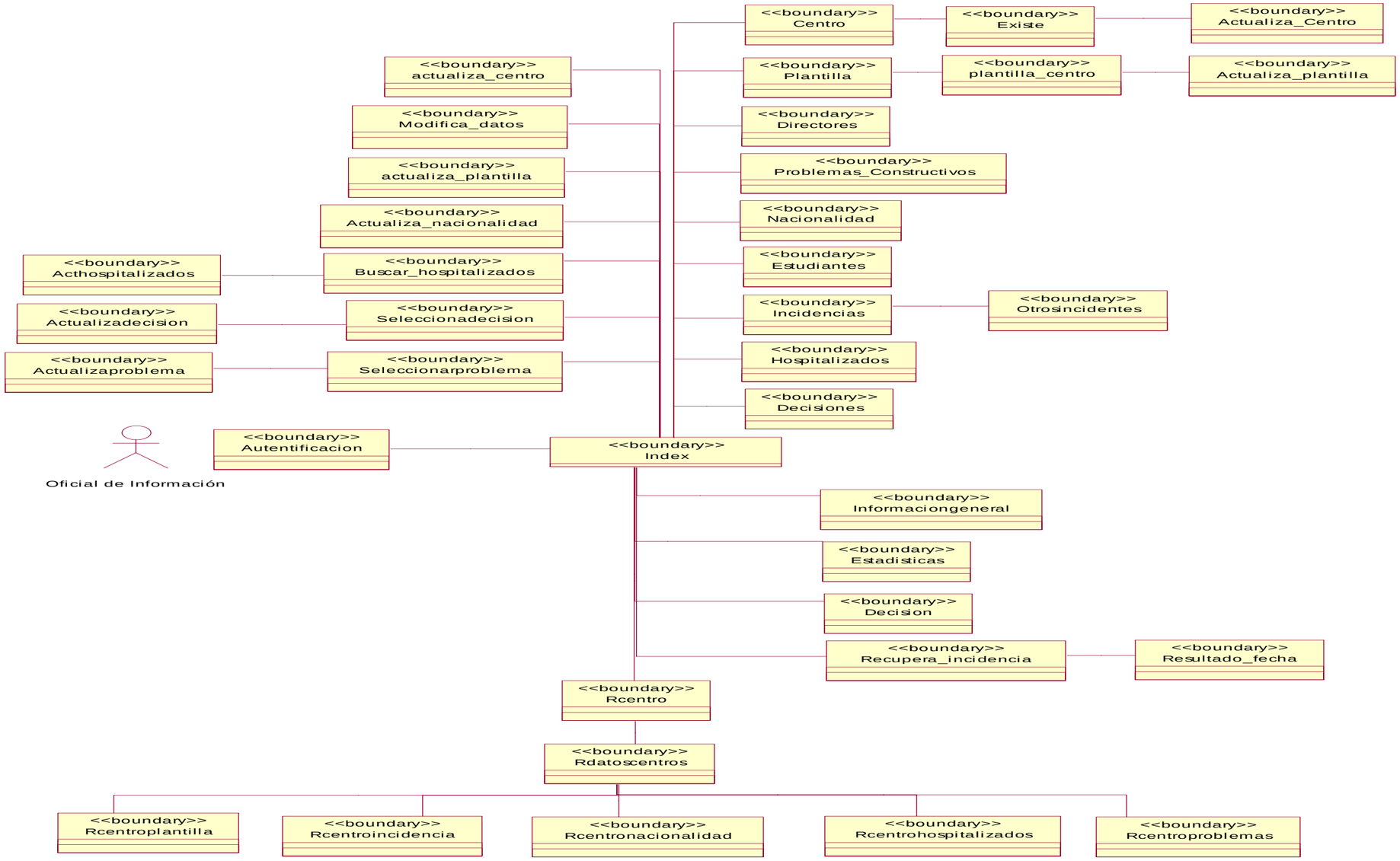


Figura 3.13 Mapa de navegación del usuario de información.

Anexo 8. Diagrama de despliegue

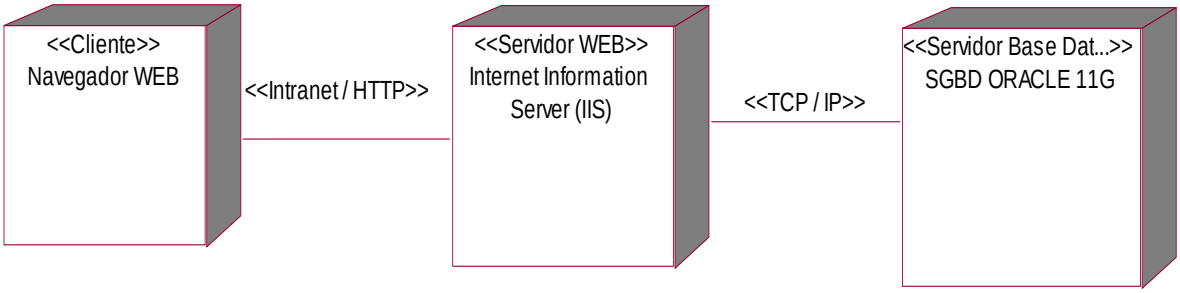


Figura 3.14 Diagrama de Despliegue.