
Análisis y Diseño de Software (ADS)

[Sistema de eventos deportivos Buenperu]

Integrantes [Luis Alexander Paredes Quichiz]
[Piero Alexander Barbaran Lagua]
[Carlos Alfredo Diaz Torrez]
[Ricardo Chimoy Correa]

[Este documento es la plantilla base para elaborar el documento Análisis y Diseño de Software. Los textos que aparecen entre paréntesis rectos son explicaciones de que debe contener cada sección. Dichos textos se deben seleccionar y sustituir por el contenido que corresponda. En caso que alguna de las secciones del presente documento no aplique a su proyecto pueden usarse las frases "No hay cambios", "No hay impacto en esta sección", "La solución que se está implementando no tiene impacto en esta sección", "No aplican para el proyecto" (No borrar secciones del documento)]

HISTORIAL DE REVISIONES

Versión	Autor	Descripción	Fecha de Elaboración	Fecha de Revisión	Revisado por
<1.1>	< <i>Luis Alexander Paredes Quichiz</i> >	<Detalles>	<06/02/26>	<Fecha de Revisión>	<Enrique Mancilla>

Contenido

1. Introducción	4
1.1. PROPÓSITO	4
1.2. ALCANCE	4
1.3. DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS	4
1.3.1. Definiciones	4
1.3.2. Acrónimos	5
1.3.3. Abreviaturas	5
1.4. REFERENCIAS.....	5
2. Modelo de Análisis.....	5
2.1. ARQUITECTURA DEL SISTEMA	5
2.2. REALIZACIÓN DE CASOS DE USO – ANÁLISIS.....	6
2.3. MODELO CONCEPTUAL	7
3. Metas y Restricciones de la Arquitectura	8
4. Modelo de Diseño	13
4.1. MODELO LÓGICO	13
4.2. MODELO FÍSICO DE DATOS	14
4.3. MODELO DE DISEÑO	15
4.3.1. Vista de Capas y Subsistemas	15
4.3.1.1. Capa de Presentación	15
4.3.1.2. Capa Controladora	15
4.3.1.3. Capa de Negocio.....	15
4.3.2. Realización de Casos de Uso – Modelo de Diseño	15
4.3.2.1. Código del CUS – Nombre del CUS 01	16
4.3.2.2. Código del CUS – Nombre del CUS 02	16
4.3.2.3. Código del CUS – Nombre del CUS 03	16
4.3.2.4. Código del CUS – Nombre del CUS 04	17
4.3.2.5. Código del CUS – Nombre del CUS 01	17
5. Vista de Procesos	18
6. Vista de Despliegue.....	18
7. Vista de Implementación	20
8. Vista de Integración del Software.....	20
8.1. CRITERIOS DE INTEGRACIÓN DE SOFTWARE	21
8.2. SECUENCIA DE INTEGRACIÓN.....	21
8.3. ENTORNO NECESARIO PARA LA INTEGRACIÓN.....	22
9. Tamaño y Desempeño	23

1. Introducción

Este documento presenta el Análisis y Diseño de Software del Sistema de Gestión de Eventos Deportivos, cuyo objetivo es automatizar la organización de torneos distritales. El sistema permitirá gestionar eventos, equipos, partidos y resultados de forma centralizada, reduciendo errores y mejorando la difusión de la información. Además, complementa la Especificación de Software e incluye los elementos necesarios para el desarrollo del sistema.

1.1. Propósito

El propósito de este documento es describir la arquitectura del sistema y detallar los diferentes modelos que la componen:

- Vista de arquitectura del software
- Vista lógica
- Vista de procesos
- Vista de implementación
- Vista de despliegue
- Modelo conceptual del dominio

Alcance

El alcance de este documento comprende el análisis y diseño del sistema desde una perspectiva arquitectónica y estructural.

Se incluyen los siguientes elementos:

Incluye:

- Arquitectura del sistema basada en capas.
- Modelo de análisis a partir de los casos de uso identificados.
- Modelo conceptual del dominio del negocio (eventos, partidos, equipos, usuarios).
- Diagramas de secuencia y clases de análisis.
- Vista de procesos, despliegue e implementación.

No incluye:

- Detalles técnicos de programación.
- Implementación del código fuente.
- Modelos de prueba o manuales de usuario

1.2. Definiciones, Acrónimos y Abreviaturas

1.2.1. Definiciones

Definición	Descripción
Organizador del torneo	Usuario encargado de crear eventos, registrar equipos, definir reglas y programar partidos.

<i>Delegado Equipo</i>	<i>Conjunto de jugadores inscritos para participar en un torneo distrital.</i>
<i>Árbitro / Planillero</i>	<i>Usuario responsable de registrar resultados oficiales de los partidos.</i>
<i>Evento Deportivo</i>	<i>Torneo o campeonato registrado en el sistema (liga, torneo simple, amistoso).</i>
<i>Partido</i>	<i>Encuentro programado entre dos equipos con un árbitro asignado.</i>
<i>Tabla de posiciones</i>	<i>Resultado acumulado del torneo basado en puntos, goles y estadísticas.</i>

1.2.2. Acrónimos

Acrónimo	Descripción
RUP	Rational Unified Process
MCU	Modelo de Casos de Uso
MAN	Modelo de Análisis de Negocio
MA	Modelo de análisis

1.2.3. Abreviaturas

Acrónimo	Descripción
SIGA	Sistema Integrado de Gestión Administrativa
UI	Interfaz de usuario

1.3. Referencias

- *Modelo de Casos de Uso del Sistema (MCU).*
- *Especificación de Software (ES) – Documento base del proyecto.*

2. Modelo de Análisis

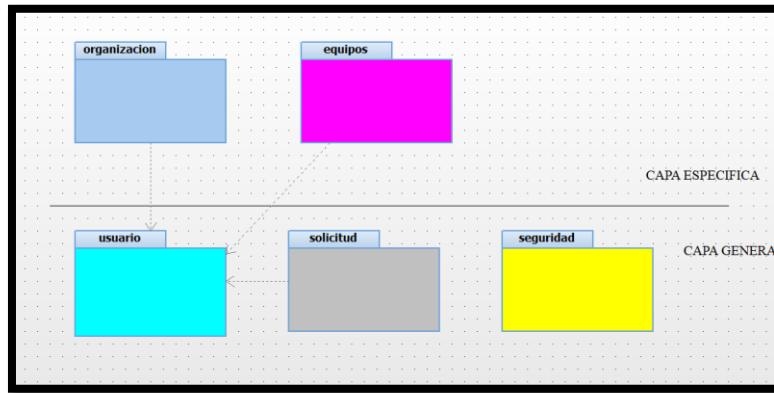
2.1. Arquitectura del Sistema

La arquitectura propuesta para el Sistema de Gestión de Eventos Deportivos está basada en el patrón MVC (Modelo-Vista-Controlador) y en una estructura modular por paquetes, permitiendo separar responsabilidades y facilitar la escalabilidad.

Paquetes del sistema:

- P01 – Usuarios:** roles, organizadores, planilleros.
- P02 – Equipos:** registro, delegados, información del equipo.

- P03 – Organización:** creación de eventos, partidos, etc.
- P04 – Seguridad:** acceso al sistema, autenticación.
- P05 – Solicitudes:** envío de solicitud, validación de solicitud.



2.2. Realización de Casos de Uso – Análisis

A partir del MCU del informe, se identifican los casos de uso principales:

- Registrar Evento
- Registrar Equipos
- Programar Partidos
- Registrar Resultados
- Consultar Tabla de Posiciones
- Autenticarse / Gestionar Usuarios

Código del CUS – Nombre del CUS

Nombre del Escenario

Caso de Uso :Mantener partidos (CUS01)

Diagrama de Clases de Análisis

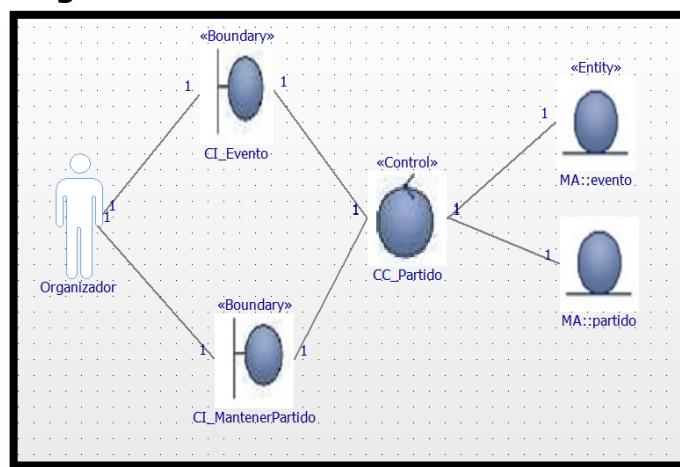


Diagrama de Secuencia

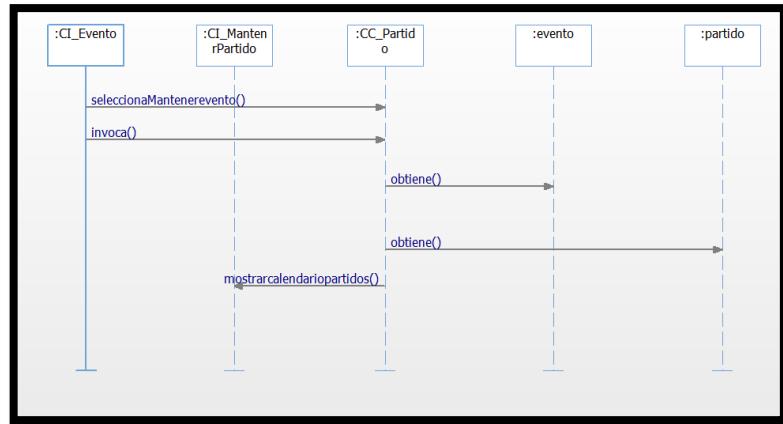
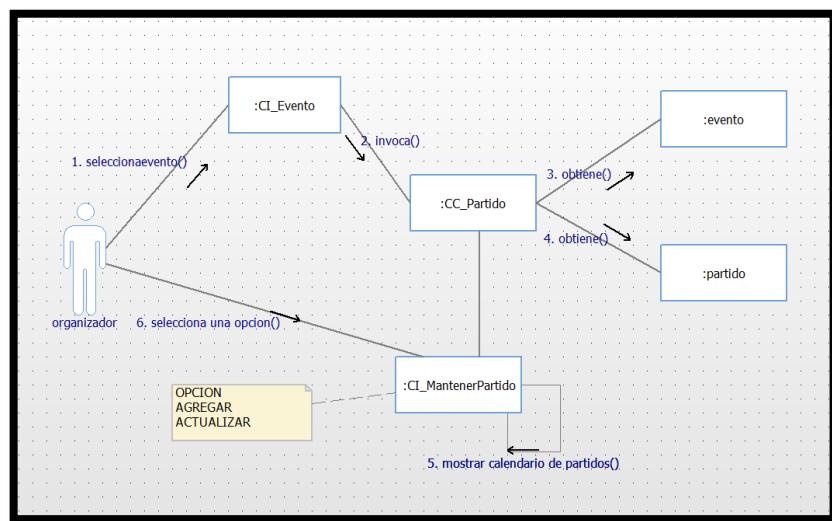
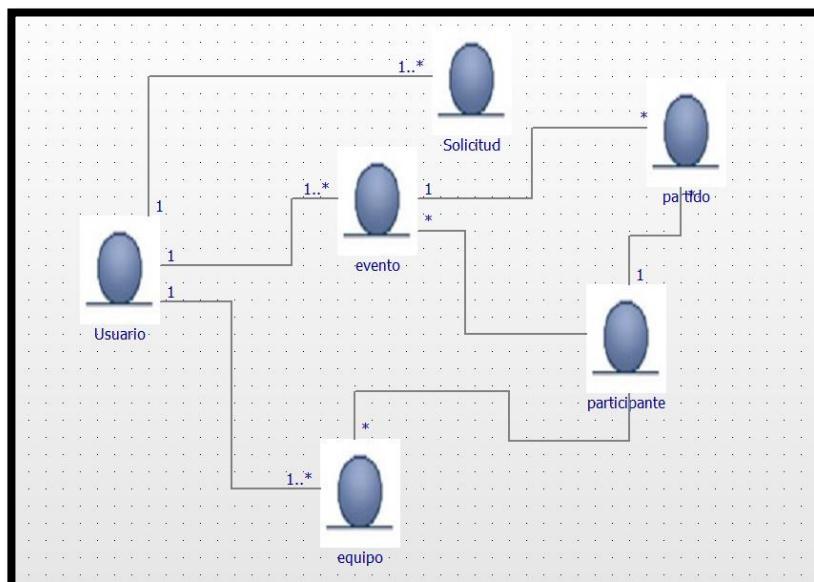


Diagrama de Comunicación



2.3. Modelo Conceptual



A continuación se presenta el diccionario de clases

Nombre	<i>Usuario</i>
Tipo	<i>Entidad</i>
Descripción	<i>Representa a administradores, organizadores y planilleros que interactúan con el sistema.</i>

Atributo	Tipo de Dato	Visibilidad	Valor inicial
<i>idUsuario</i>	<i>Integer</i>	<i>Privado</i>	—
<i>nombres</i>	<i>String</i>	<i>Privado</i>	—
<i>apellidos</i>	<i>String</i>	<i>Privado</i>	—
<i>email</i>	<i>String</i>	<i>Privado</i>	—
<i>password</i>	<i>String</i>	<i>Privado</i>	—
<i>distrito</i>	<i>String</i>	<i>Privado</i>	—
<i>telefono</i>	<i>String</i>	<i>Privado</i>	—
<i>rol</i>	<i>Enum(admin, organizer, payroll)</i>	<i>Privado</i>	<i>organizer</i>
<i>estado</i>	<i>Enum(active, inactive, banned)</i>	<i>Privado</i>	<i>active</i>

1.

Nombre	<i>Solicitud</i>
Tipo	<i>Entidad</i>
Descripción	<i>Solicitud enviada por un usuario para validación/habilitación (p. ej., organizador).</i>

Atributo	Tipo de Dato	Visibilidad	Valor inicial
<i>idSolicitud</i>	<i>Integer</i>	<i>Privado</i>	—
<i>idUsuario</i>	<i>Integer</i>	<i>Privado</i>	—
<i>archivoSolicitud</i>	<i>Binary</i>	<i>Privado</i>	—
<i>emailOrganizador</i>	<i>String</i>	<i>Privado</i>	—
<i>estado</i>	<i>Enum(pending, approved, rejected)</i>	<i>Privado</i>	<i>pending</i>
<i>fechaCreacion</i>	<i>DateTime</i>	<i>Privado</i>	<i>CURRENT_TIMESTAMP</i>
<i>fechaRevision</i>	<i>DateTime</i>	<i>Privado</i>	—
<i>revisadoPor</i>	<i>Integer</i>	<i>Privado</i>	—

2.

Nombre	<i>Evento</i>
Tipo	<i>Entidad</i>
Descripción	<i>Torneo o campeonato registrado en el sistema (liga, torneo, amistoso) con reglas de puntuación y configuración.</i>

Atributo	Tipo de Dato	Visibilidad	Valor inicial
<i>idEvento</i>	<i>Integer</i>	<i>Privado</i>	—
<i>nombre</i>	<i>String</i>	<i>Privado</i>	—
<i>fuenteImagen</i>	<i>String</i>	<i>Privado</i>	—
<i>tipoEvento</i>	<i>Enum(league, tournament, friendly)</i>	<i>Privado</i>	—

<i>tipoDeporte</i>	<i>String</i>	<i>Privado</i>	—
<i>distrito</i>	<i>String</i>	<i>Privado</i>	—
<i>categoria</i>	<i>String</i>	<i>Privado</i>	—
<i>fechaInicio</i>	<i>Date</i>	<i>Privado</i>	—
<i>fechaFin</i>	<i>Date</i>	<i>Privado</i>	—
<i>puntosVictoria</i>	<i>Integer</i>	<i>Privado</i>	<i>3</i>
<i>puntosEmpate</i>	<i>Integer</i>	<i>Privado</i>	<i>1</i>
<i>puntosDerrota</i>	<i>Integer</i>	<i>Privado</i>	<i>0</i>
<i>maxEquipos</i>	<i>Integer</i>	<i>Privado</i>	—
<i>organizadorId</i>	<i>Integer</i>	<i>Privado</i>	—
<i>estado</i>	<i>Enum(draft, active, finished)</i>	<i>Privado</i>	<i>draft</i>

3.

Nombre	<i>Equipo</i>
Tipo	<i>Entidad</i>
Descripción	<i>Equipo participante de eventos deportivos; puede estar asociado a un usuario (delegado/creador).</i>

Atributo	Tipo de Dato	Visibilidad	Valor inicial
<i>idEquipo</i>	<i>Integer</i>	<i>Privado</i>	—
<i>nombre</i>	<i>String</i>	<i>Privado</i>	—
<i>distrito</i>	<i>String</i>	<i>Privado</i>	—
<i>categoria</i>	<i>String</i>	<i>Privado</i>	—
<i>icono</i>	<i>String</i>	<i>Privado</i>	—
<i>creadorId</i>	<i>Integer</i>	<i>Privado</i>	—
<i>activo</i>	<i>Boolean</i>	<i>Privado</i>	<i>true</i>

4.

Nombre	<i>ParticipacionEquipo</i>
Tipo	<i>Entidad (Clase de enlace)</i>
Descripción	<i>Inscripción de un equipo en un evento, con estadísticas acumuladas para tabla de posiciones.</i>

Atributo	Tipo de Dato	Visibilidad	Valor inicial
<i>idParticipacion</i>	<i>Integer</i>	<i>Privado</i>	—
<i>idEvento</i>	<i>Integer</i>	<i>Privado</i>	—
<i>idEquipo</i>	<i>Integer</i>	<i>Privado</i>	—
<i>puntos</i>	<i>Integer</i>	<i>Privado</i>	<i>0</i>
<i>posicionFinal</i>	<i>Integer</i>	<i>Privado</i>	—
<i>partidosJugados</i>	<i>Integer</i>	<i>Privado</i>	<i>0</i>
<i>victorias</i>	<i>Integer</i>	<i>Privado</i>	<i>0</i>
<i>empates</i>	<i>Integer</i>	<i>Privado</i>	<i>0</i>
<i>derrotas</i>	<i>Integer</i>	<i>Privado</i>	<i>0</i>

5.

Nombre	<i>Partido</i>
Tipo	<i>Entidad</i>
Descripción	<i>Encuentro programado dentro de un evento entre dos participaciones (equipos inscritos), con árbitro y</i>

	<i>marcador.</i>
--	------------------

Atributo	Tipo de Dato	Visibilidad	Valor inicial
<i>idPartido</i>	<i>Integer</i>	<i>Privado</i>	—
<i>idEvento</i>	<i>Integer</i>	<i>Privado</i>	—
<i>fechaHora</i>	<i>DateTime</i>	<i>Privado</i>	—
<i>lugar</i>	<i>String</i>	<i>Privado</i>	—
<i>estado</i>	<i>Enum(scheduled, in_progress, completed, canceled)</i>	<i>Privado</i>	<i>scheduled</i>
<i>arbitro</i>	<i>String</i>	<i>Privado</i>	—
<i>ronda</i>	<i>String</i>	<i>Privado</i>	—
<i>participacionLocalId</i>	<i>Integer</i>	<i>Privado</i>	—
<i>participacionVisitanteId</i>	<i>Integer</i>	<i>Privado</i>	—
<i>scoreLocal</i>	<i>Integer</i>	<i>Privado</i>	0
<i>scoreVisitante</i>	<i>Integer</i>	<i>Privado</i>	0

3. Metas y Restricciones de la Arquitectura

[En ésta sección se describen los requerimientos de software y objetivos que tienen algún significativo impacto sobre la arquitectura; por ejemplo: seguridad, privacidad de uso del producto, portabilidad, distribución y reuso. Esto también captura las restricciones especiales que quizás apliquen en la: Estrategia de diseño e implementación, herramientas de desarrollo, estructura del equipo, cronograma, leyes y regulaciones legales y otros. Las restricciones que aquí se recogen pueden complementar a las identificadas en el ES a excepción de aquellas funcionales.]

Ejemplo:

A continuación se presenta el listado de requerimientos que tienen impacto sobre la arquitectura del módulo de Programación de Viajes:

Clasificación	Descripción	Requerimientos
Usabilidad	Se enfoca a las características de estética y consistencia en las interfaces gráficas	<p>RNF03 – El sistema deberá permitir a los usuarios realizar consultas según su perfil de acceso.</p> <p>RNF04 – La interfaz del usuario se diseñará de tal manera que le facilite el uso de la misma, sin necesidad de un soporte del área de sistemas. Esta tendrá que ser validada por el usuario.</p> <p>RNF05 – En caso de error del usuario el sistema informará claramente: el mensaje del error y la solución.</p> <p>RNF06 – El lenguaje empleado en la interfaz gráfica del sistema respetará los términos usados en el negocio.</p>

Clasificación	Descripción	Requerimientos
		<p>RNF31- El sistema tendrá un menú de ayuda donde se encuentran especificadas el funcionamiento de las principales funciones del sistema.</p> <p>RNF33 – La resolución mínima para una buena visualización y ejecución del sistema será un tamaño de pantalla de 800x600 píxel.</p> <p>RNF34 – Formato del menú del sistema.</p>
Confiabilidad	Se enfoca con las características como disponibilidad (el tiempo disponible del sistema), exactitud de los cálculos del sistema, y las habilidades del sistema para recuperarse durante fallos.	<p>RNF08 – El sistema estará disponible 24 horas al día, 7 días a la semana.</p> <p>RNF09 – El porcentaje de disponibilidad anual del sistema no será menor del 99%.</p> <p>RNF10 – El tiempo promedio entre fallas estimado será de una vez cada 6 semanas</p> <p>RNF11 – El sistema deberá mantener almacenado los errores originados por excepciones en el sistema.</p> <p>RNF12 – El sistema deberá mantener almacenado el contenido histórico de todas las operaciones (Log).</p> <p>RNF13 – El tiempo promedio de corrección del sistema no debe superar las 4 horas.</p>
Rendimiento	Se enfoca con las características como tiempo de respuesta, tiempo de iniciación y término.	<p>RNF15 – El Sistema deberá permitir el ingreso concurrente de por lo menos 200 usuarios distribuidos entre los diversos módulos del sistema a lo largo de las diversas oficinas con las que cuentan la compañía.</p> <p>RNF16 – El tiempo de respuesta del sistema para operaciones de ingreso o registro de información deberá ser como máximo 5 segundos de espera.</p>

Clasificación	Descripción	Requerimientos
		<p>RNF17 – El tiempo promedio de cada transacción realizada en el sistema deberá ser en promedio de 8 segundos.</p> <p>RNF18 – El tiempo promedio de cada consulta realizada en el sistema deberá ser menor a 10 segundos.</p> <p>RNF19 – El sistema deberá soportar un promedio de 50 transacciones por minuto.</p> <p>RNF20 – El tiempo de carga de pantalla deberá ser en promedio 5 segundos</p> <p>RNF21 – El sistema deberá permitir como máximo 10 pantallas abiertas por usuario.</p>
Soporte	Se concentra en las características como pruebas, adaptabilidad, mantenimiento, configuración, Instalación, escalabilidad, y localización.	<p>RNF22 – El sistema será compatible con Windows 2000 profesional y Windows XP profesional o superiores.</p> <p>RNF24 – El sistema deberá mostrar la versión correspondiente en la opción del menú "Acerca de".</p>
Consideraciones de diseño	Especifica las opciones del diseño para el sistema.	<p>RNF26 – El sistema debe alinearse con la red implementada en la empresa y no deberá generar conflicto con las aplicaciones existentes.</p> <p>RNF27 – El sistema debe trabajar sobre cualquier computador que cuente con estos requerimientos mínimos con procesador Pentium III o superior, 256 Mb de memoria RAM y disco duro de 20 Gb.</p> <p>RNF28 – La aplicación se desarrollará con la herramienta Visual Basic versión 6 de Microsoft.</p> <p>RNF29 – El sistema deberá considerar una arquitectura lógica de tres capas: Datos, Negocio y</p>

Clasificación	Descripción	Requerimientos
		<p>Presentación.</p> <p>RNF30 – El motor de base de datos que utilizará el sistema deberá ser SQL Server 2000.</p> <p>RNF28 – La aplicación se desarrollará con la herramienta Visual Basic versión 6 de Microsoft.</p>
Requerimientos de implementación	Especifica la codificación o construcción del sistema, pueden ser estándares, implementaciones, lenguajes y límites de los recursos.	RNF28 – La aplicación se desarrollará con la herramienta Visual Basic versión 6 de Microsoft.
Requerimiento físicos	Especificaciones físicas impuestas por el hardware usado para mantener el sistema.	RNF27- Las cuales en las cuales se instalará las aplicaciones deberán ser Pentium III 700 Ghz. o superior. Las cuales deberán tener disponible como mínimo 200MB libres de disco y deberán contar con un mínimo de 128MB de RAM.
Aspectos Generales	Especifica los requerimientos de seguridad que deben tener el sistema y sus características generales.	[RNF35] - A cada usuario se le asignará un usuario del sistema y una clave, los cuales permitirán el ingreso de acuerdo un perfil determinado

4. Modelo de Diseño

4.1. Modelo Lógico

*[El modelo lógico es el **refinamiento del Modelo Conceptual**. Aquí se reducen y/o aumentan clases y sólo quedan aquellas que van a ser diseñadas como tablas de la Base de Datos. El modelo lógico debe representarse con un diagrama de clases de acuerdo a la arquitectura propuesta. Tenga presente que para la transformación del modelo conceptual al modelo lógico se debe tener en cuenta:*

- *Pasar las reglas de negocio*
- *Colocar las multiplicidades entre clases*
- *Identificar los atributos de Enlace o Clase de Enlace de las asociaciones de muchos a muchos*
- *NO INCLUIR los atributos identificadores de la clase (se agregarán en el modelo físico)*

- Incluir los atributos de las clases que se necesitan para satisfacer los requerimientos del sistema
- Documentar un registro de glosario de términos
- Verificar que las reglas de negocio se sigan cumpliendo.

[Se sugiere que por cada clase se tenga un diccionario que incluya el nombre, el tipo, la descripción, atributos, tipo de dato, visibilidad y valor inicial]

Nombre	<i>Nombre de la Clase</i>		
Tipo	<i>Tipo de Clase (Ejemplo Entidad)</i>		
Descripción	<i>Descripción de la clase identificando que representa</i>		
Atributo	Tipo de Dato	Visibilidad	Valor inicial
Nombre del atributo	Integer / String / Boolean	Público / Privado	

4.2. Modelo Físico de datos

[El modelo Físico es la estructura final de la base de datos, aquí se debe de considerar todos las tablas necesarias para cumplir las reglas de seguridad y auditoria .así como el diccionario de datos.]

Diccionario de Datos

Tabla:			
Descripción:			
Campo	Tipo	Longitud	Descripción
Restricciones:			
Llaves Primarias:			
Llaves Foráneas:			

Ejemplo

Tabla:	TB_Producto		
Descripción:	Contiene los datos de los productos que comercializa el negocio.		
Campo	Tipo	Longitud	Descripción
Pro_Codigo	Carácter	6	Identificador único del producto
Pro_Nombre	Carácter	60	Descripción del nombre del producto
Pro_Stock	Numérico	8	Cantidad actual del producto
Pro_Precio1	Numérico	8	Precio Venta 1 del producto
Pro_Precio2	Numérico	8	Precio Venta 2 del producto
Pro_Precio3	Numérico	8	Precio Venta 3 del producto
Pro_Precio4	Numérico	8	Precio Venta 4 del producto

Pro_Precio5	Numérico	8	Precio Venta 5 del producto
Pro_Estado	Carácter	1	Estado del producto 1=Activo 2=Inactivo
Pro_Envase	Carácter	1	Tipo de envase del producto 1=Lata, 2=Caja, 3=Bolsa,4=paquete
Lin_Codigo	Carácter	2	identificador único de la línea
Restricciones:	El campo Pro_codigo es único. El campo Pro_Nombre no puede ser nulo. El campo Pro_Envase se asigna por defecto el valor 4 El campo Pro_Estado se asigna por defecto el valor 1		
Llaves Primarias:	Pro_Codigo		
Llaves Foráneas:	Lin_Codigo		

4.3. Modelo de Diseño

[En esta sección debe representar el refinamiento del modelo de análisis considerando los requisitos no funcionales identificados en la ES.]

4.3.1. Vista de Capas y Subsistemas

[Incluir el diagrama en el que se represente la arquitectura de diseño. Para ello puede usar un patrón en el cual se usen capas y subsistemas. Además deberá identificar subsistemas requeridos por el uso de algún patrón de diseño como el DAO Factory, Singleton, Front Controller, entre otros. Por cada capa y subsistema deberá identificar las clases de diseño que se implementarán]

4.3.1.1. Capa de Presentación

[Identifique las clases de diseño de la capa de presentación. Ordene dicha identificación utilizando los paquetes al interior de las capas denominados subsistemas.]

4.3.1.2. Capa Controladora

[Identifique las clases de diseño de la controladora. Ordene dicha identificación utilizando los paquetes al interior de las capas denominados subsistemas.]

4.3.1.3. Capa de Negocio

[Identifique las clases de diseño de la capa de negocio. Ordene dicha identificación utilizando los paquetes al interior de las capas denominados subsistemas.]

4.3.2. Realización de Casos de Uso – Modelo de Diseño

[Esta sección deberá desarrollar los diagramas de secuencia y de clases de diseño a partir de los requisitos funcionales identificados en la ES y considerando los escenarios identificados del presente documento. Debe asegurarse que

[las clases que se incorporen deben ser aquellas que se han identificado del presente documento.]

4.3.2.1. Código del CUS – Nombre del CUS 01

[A partir de los casos de uso realizados del modelo de análisis deberá identificar los casos de uso que usará para las realizaciones de diseño.]

Nombre del Escenario

[Identifica el escenario a ser realizado y una breve descripción. Se recomienda identificar con un código único a cada escenario. Por ejemplo ESC01. Deberá reusar los escenarios identificados en el modelo de análisis]

Diagrama de Secuencia de Diseño

[Incluya el diagrama de secuencia de diseño en el cual se observe el uso de patrones de diseño para las clases que implementarán cada una de las clases lógicas.]

Diagrama de Clases de Diseño

[Incluya el diagrama de clases de diseño obtenido del conjunto de diagramas de secuencia que se implementan por cada escenario.]

4.3.2.2. Código del CUS – Nombre del CUS 02

[A partir de los casos de uso realizados del modelo de análisis deberá identificar los casos de uso que usará para las realizaciones de diseño.]

Nombre del Escenario

[Identifica el escenario a ser realizado y una breve descripción. Se recomienda identificar con un código único a cada escenario. Por ejemplo ESC01. Deberá reusar los escenarios identificados en el modelo de análisis]

Diagrama de Secuencia de Diseño

[Incluya el diagrama de secuencia de diseño en el cual se observe el uso de patrones de diseño para las clases que implementarán cada una de las clases lógicas.]

Diagrama de Clases de Diseño

[Incluya el diagrama de clases de diseño obtenido del conjunto de diagramas de secuencia que se implementan por cada escenario.]

4.3.2.3. Código del CUS – Nombre del CUS 03

[A partir de los casos de uso realizados del modelo de análisis deberá identificar los casos de uso que usará para las realizaciones de diseño.]

Nombre del Escenario

[Identifica el escenario a ser realizado y una breve descripción. Se recomienda identificar con un código único a cada escenario. Por ejemplo ESC01. Deberá reusar los escenarios identificados en el modelo de análisis]

Diagrama de Secuencia de Diseño

[Incluya el diagrama de secuencia de diseño en el cual se observe el uso de patrones de diseño para las clases que implementarán cada una de las clases lógicas.]

Diagrama de Clases de Diseño

[Incluya el diagrama de clases de diseño obtenido del conjunto de diagramas de secuencia que se implementan por cada escenario.]

4.3.2.4. Código del CUS – Nombre del CUS 04

[A partir de los casos de uso realizados del modelo de análisis deberá identificar los casos de uso que usará para las realizaciones de diseño.]

Nombre del Escenario

[Identifica el escenario a ser realizado y una breve descripción. Se recomienda identificar con un código único a cada escenario. Por ejemplo ESC01. Deberá reusar los escenarios identificados en el modelo de análisis]

Diagrama de Secuencia de Diseño

[Incluya el diagrama de secuencia de diseño en el cual se observe el uso de patrones de diseño para las clases que implementarán cada una de las clases lógicas.]

Diagrama de Clases de Diseño

[Incluya el diagrama de clases de diseño obtenido del conjunto de diagramas de secuencia que se implementan por cada escenario.]

4.3.2.5. Código del CUS – Nombre del CUS 01

[A partir de los casos de uso realizados del modelo de análisis deberá identificar los casos de uso que usará para las realizaciones de diseño.]

Nombre del Escenario

[Identifica el escenario a ser realizado y una breve descripción. Se recomienda identificar con un código único a cada escenario. Por ejemplo ESC01. Deberá reusar los escenarios identificados en el modelo de análisis]

Diagrama de Secuencia de Diseño

[Incluya el diagrama de secuencia de diseño en el cual se observe el uso de patrones de diseño para las clases que implementarán cada una de las clases lógicas.]

Diagrama de Clases de Diseño

[Incluya el diagrama de clases de diseño obtenido del conjunto de diagramas de secuencia que se implementan por cada escenario.]

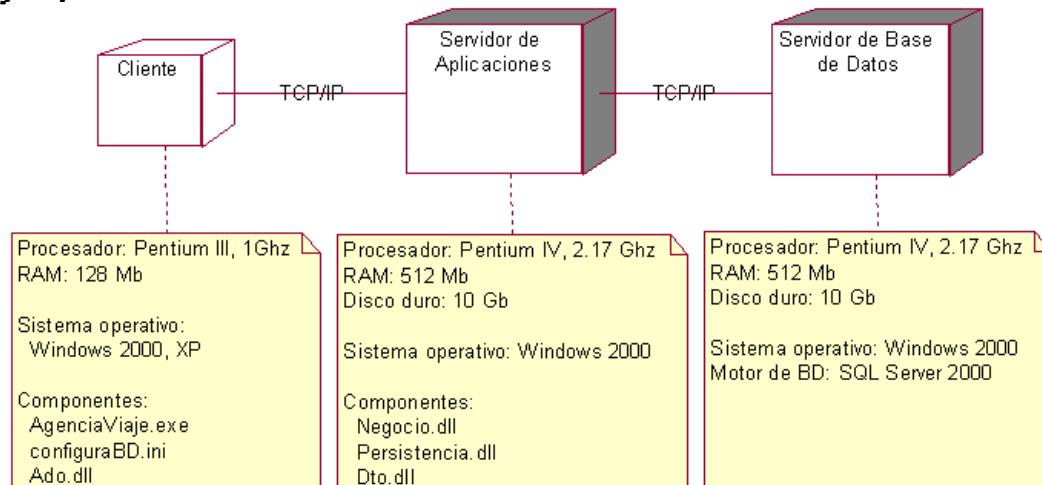
5. Vista de Procesos

[Esta sección describe la descomposición del sistema en procesos de primer nivel (un solo hilo de control) y los procesos de último nivel (grupos de procesos de primer nivel). También describe la ubicación de objetos y clases. Organizar la sección por los grupos de los procesos que se comunican u obran recíprocamente. Describir los modos principales de la comunicación entre los procesos, tales como el paso de mensajes, interrupciones y qué pasa, las interrupciones, y puntos de encuentro entre procesos.]

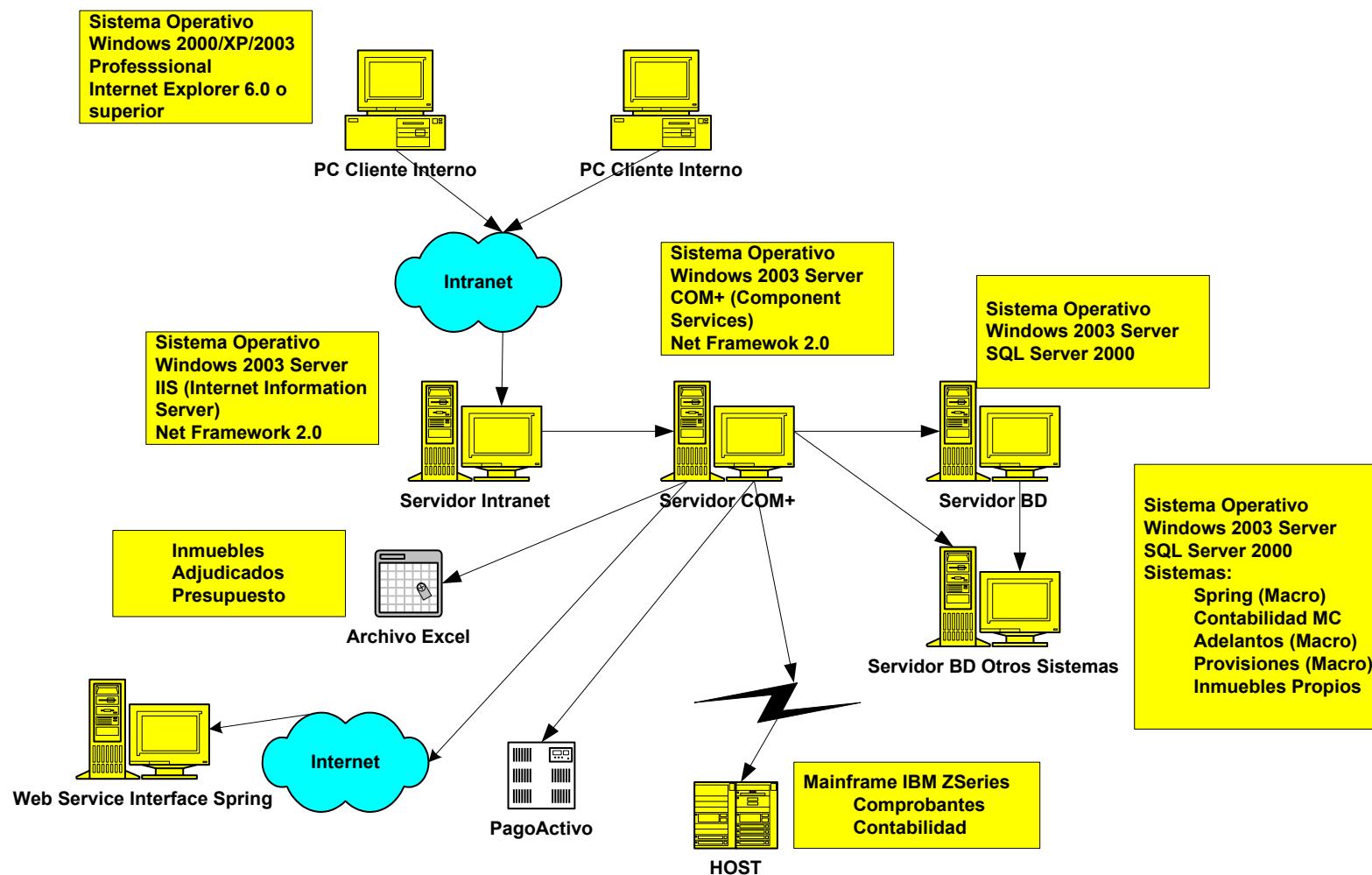
6. Vista de Despliegue

[En esta sección se describen unas o más configuraciones físicas de la red (hardware) que se usarán para el despliegue de los componentes de software que forman parte de la solución. Para ello puede usar un Diagrama de Despliegue indicando como mínimo, para cada configuración, en qué nodos físicos (computadoras, CPU) se ejecuta el software y sus interconexiones (bus, LAN, punto a punto, y así sucesivamente). De ser posible se debe incluir un mapeo de los procesos de la vista de procesos sobre los nodos físicos. Además deberá especificar los detalles técnicos de cada nodo en la vista de despliegue.]

Ejemplo 1:



Ejemplo 2:



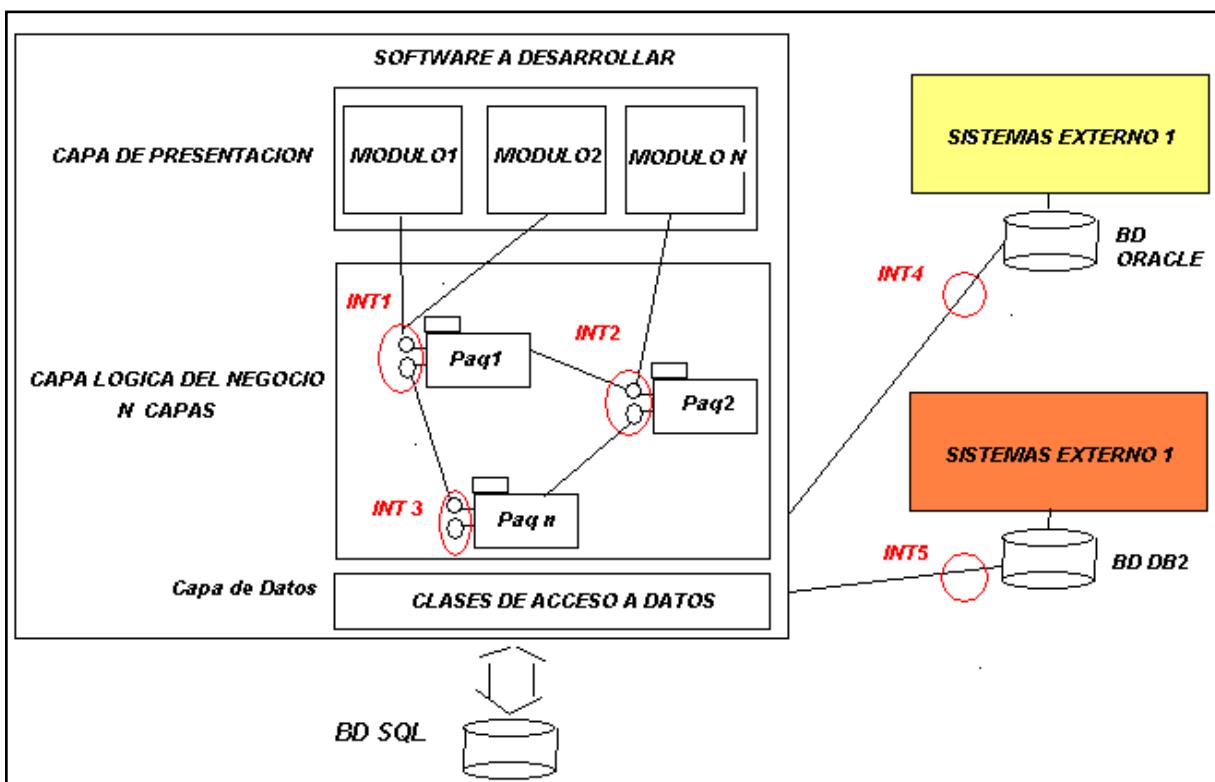
7. Vista de Implementación

[En esta sección se describe la estructura total del modelo de implementación, utilizando la descomposición del software en capas y subsistemas y cómo éste se pondrá en práctica. Deberá identificar cualquier componente arquitectónico significativo. Debe nombrar y definir las capas y contenidos de las mismas, las reglas que gobiernan la inclusión de una u otra capa, y las características entre capas. Incluya el diagrama de componentes que muestra las relaciones entre capas. Para cada capa, incluya una sub-sección con el nombre de la capa, una enumeración de los subsistemas localizados dentro de la capa y un diagrama de componentes.]

8. Vista de Integración del Software

[De requerirlo en esta sección puede incluir un diagrama integración del software desarrollado y su interacción con las diferentes interfaces identificadas en el modelo de diseño.]

Ejemplo:



INTERFAZ	DESCRIPCION BREVE	TIPO DE INTERFAZ	REFERENCIA
INT1	La interfaz 1 apoya la integración del Paquete 1 y el Paquete 2, incluye las clases C1, C2, etc....	Interfaz Interna	La Especificación de esta interfaz se encuentra en el documento de Especificación de Componentes
INT2	La interfaz 1 apoya la integración del Paquete 1 y el Paquete 2, incluye las	Interfaz Interna	La Especificación de esta interfaz se encuentra en el documento de

INTERFAZ	DESCRIPCION BREVE	TIPO DE INTERFAZ	REFERENCIA
	clases C1, C2, etc....		Especificación de Componentes
INT3		Interfaz Interna	La Especificación de esta interfaz se encuentra en el documento de Especificación de Componentes
INT4		Interfaz Externa	La Especificación de esta interfaz se encuentra en el documento de Especificación de Componentes
INT5		Interfaz Externa	La Especificación de esta interfaz se encuentra en el documento de Especificación de Componentes

8.1. Criterios de Integración de Software

[Identifique los criterios que se deberán considerar para la integración de los componentes de software y sus interfaces.

Ejemplo:

Para la óptima integración del Software se deberán tener que cumplir, considerar y evaluar los siguientes criterios:

- *Antes de realizar la integración todos los componentes deberán haber pasado por pruebas unitarias.*
- *Antes de realizar la integración, todas las incidencias, errores u otras no conformidades encontradas durante las pruebas unitarias deberán estar cerradas.*
- *Se deberá tener preparado los ambientes y entornos para la integración (Entorno de Desarrollo o Entorno de Integración).*
- *Deberá haberse inicializado y migrado data consistente previa a la integración.*
- *Otros Criterios que apoyen a que la integración resulte un éxito.]*

8.2. Secuencia de Integración

[Defina la secuencia de integración que se aplicarán a los componentes de software y sus interfaces.

Ejemplo:

Para que el Software se integre totalmente se seguirá la siguiente secuencia de integración:

- *Realizar las pruebas unitarias a todos los componentes desarrollados (De todos los módulos).*
- *Levantar todos los errores e incidencias encontradas en las pruebas unitarias (De todos los módulos).*
- *Realizar revisión de pares al código fuente y levantar las no conformidades.*

- Asegurarse que todos los componentes del Sistema estén completamente corregidos (Realización de nuevas pruebas sobre los errores encontrados).
- Validar que el entorno de integración este listo.
- Validar que la data haya sido migrada satisfactoriamente.
- Iniciar la integración
 - Integrar Modulo 1 y Modulo 2 - Realizar pruebas de integración entre ambos módulos.
 - Integrar Modulo 1 y Modulo 2 y Modulo3 - Realizar pruebas de integración entre módulos.
 - Integrar Modulo 1 y Modulo 2 y Modulo n - Realizar pruebas de integración entre módulos.
- Finalizada la Integración entre módulos, realizar la integración con aplicativos externos al sistema en desarrollo.
 - Integrar Sistema en desarrollo con Sistema Externo1 (Aplicativo Externo) y Realizar Pruebas.
 - Integrar Sistema en desarrollo con Sistema Externo2 (Aplicativo Externo) y Realizar Pruebas.
- Finalmente realizar las pruebas del Sistema y luego de ellas las Pruebas de Aceptación con los Usuarios Finales.

8.3. Entorno Necesario para la Integración

[En esta sección se deberán identificar y especificar los diversos entornos que se usarán o que están involucrados en la integración del Software.]

NOMBRE DEL SERVIDOR	Serv_Desa							
IP	1.1.15.50							
DESCRIPCION Y OBJETIVO DEL SERVIDOR								
<i>En este servidor se almacenará el código fuente, en este entorno trabajarán los desarrolladores. Aquí se realizarán las pruebas unitarias.</i>								
SERVICIOS								
NOMBRE DE SERVICIO	APLICACIÓN	FUNCIÓN	INICIO	USUARIO				
<i>Por Ejemplo: Asynchronous JavaScript + XML</i>	<i>Por ejemplo: AJAX</i>	<i>Por ejemplo: Presentación basada en estándares usando XHTML y CSS</i>	<i>Automático</i>	<i>Adminservice</i>				
<i><Servicio 1></i>	<i><Aplicación 1></i>	<i><Función 1></i>	<i>Automático</i>	<i>Local System Account</i>				
<i><Servicio 2></i>	<i><Aplicación 2></i>	<i><Función 2></i>	<i>Automático</i>	<i>Local System Account</i>				
<i><Servicio N></i>	<i><Aplicación N></i>	<i><Función 1></i>	<i>Automático</i>	<i>Local System</i>				

		<i>Account</i>
CONFIGURACIÓN DE HARDWARE Y SOFTWARE		
Nombre del Sistema Operativo	<i>Microsoft (R) Windows (R) Server 200. Enterprise Edition</i>	
Version	<i>2.2.3790 Service Pack 2 Build 3790</i>	
Proveedor del Sistema Operativo	<i>Microsoft Corporation</i>	
Nombre del Sistema	<i>DEIPSBATCH</i>	
Proveedor del Sistema	<i>IBM</i>	
Modelo del Sistema	<i>-[865811Y]-</i>	
Tipo del Sistema	<i>X86 – based PC</i>	
Procesador	<i>x86 Family 6 Model 8 Stepping 3 Genuineintel - 664</i>	
BIOS Version/Date	<i>IBM ILKT44AUS, 20/09/2001</i>	
SMBIOS Version	<i>2.1</i>	
Total de Memoria Física	<i>2,047.49 MB</i>	
Promedio de Memoria Física	<i>1.37 GB</i>	
Total de Memoria Virtual	<i>3.86 GB</i>	
Promedio de Memoria Virtual	<i>3.47 GB</i>	
Tipo de Adaptador	<i>Ethernet 802.3</i>	
Tipo de Producto	<i>IBM Netfinity Fault Tolerante PCI Adapter</i>	
Nombre del Servicio	<i>PCNet5</i>	
Dirección IP	<i>10.203.32.9</i>	
Máscara de Sub Red IP	<i>255.255.255.0</i>	
Gateway IP	<i>10.203.32.254</i>	
DHCP Enabled	<i>No</i>	
DHCP Server	<i>Not Available</i>	
MAC Address	<i>00:06:29:D5:38:0F</i>	
Memory Address	<i>0XFEB7FC00-0XFEB7FC1F</i>	
<u>SOFTWARE ADICIONAL</u>		
<u>USARIOS CON PERMISOS AL SERVIDOR</u>		
<u>RELACION CON OTROS SERVIDORES</u>		

9. Tamaño y Desempeño

[En esta sección se pueden incluir descripciones de las principales características del dimensionamiento del software que afectan la arquitectura, así como las restricciones de desempeño. Si trabaja estas características en la ES haga referencia a dicho documento.]