

Instructivo para la elaboración de un Manual Técnico

Índice

1. Introducción	3
2. Confección	3
2.1 Nombre del Sistema.....	3
2.2 Versión del Sistema	3
2.3 Tipo de Manual	3
2.4 Poner una Imagen.....	3
2.5 Fecha de Elaboración	3
2.6 Área donde fue elaborado	3
2.7 Índice del Contenido del Manual	4
2.7.1. <i>Introducción</i>	4
2.7.2. <i>Objetivos Generales y Específicos del Sistema</i>	4
2.7.3. <i>Normas, Políticas y Procedimientos</i>	4
2.7.4. <i>Definición de las Reglas del Negocio Implementadas</i>	4
2.7.5. <i>Fundamentación de la Tecnología Utilizada</i>	5
2.7.6. <i>Descripción de los Actores del Sistema</i>	5
2.7.7. <i>Especificación de Requisitos</i>	5
2.7.8. <i>Vista Funcional</i>	6
2.7.9 <i>Vista Lógica</i>	8
2.7.10. <i>Modelo Lógico de Datos</i>	10
2.7.11. <i>Modelo Físico de Datos</i>	11
2.7.13. <i>Diseño de Pantallas y Reportes</i>	11
2.7.14. <i>Descripción de Campos Requeridos por Pantalla</i>	11
2.7.15. <i>Vista de Implementación</i>	11
2.7.16. <i>Vista de Despliegue</i>	12
2.7.17. <i>Diagrama de Navegación del Sistema</i>	12
2.7.18. <i>Controles de Auditoría Implementados en el Sistema</i>	12
2.7.19. <i>Glosario de Términos</i>	12
3. Estándares de Elaboración del Manual	13

1. Introducción

El Manual Técnico, como su nombre lo indica, contiene las especificaciones técnicas más importantes del sistema desarrollado. Constituye una guía especializada para la realización de las operaciones de mantenimiento de la aplicación. Se encuentra dirigido fundamentalmente a la dirección de Tecnologías de la Información, al administrador del sistema, a otros desarrolladores, así como al departamento de calidad y auditoría de sistemas.

2. Confección

Para la elaboración de un manual de técnico se deben de considerar los siguientes apartados normativos.

Nota: Los acápites que aparecen en color rojo y con descripción en letra cursiva son de carácter opcional.

2.1 Nombre del Sistema: Nombre del sistema al que se refiere el manual.

2.2 Versión del Sistema: La versión del sistema en el manual nos permitirá mantener un control sobre las modificaciones que han afectado al sistema original.

2.3 Tipo de Manual: Se especifica el tipo de manual al que se hace referencia, permitiendo tener un control en nuestros manuales, además de una fácil identificación.

2.4 Poner una Imagen: Es recomendable ilustrar el manual con una imagen representativa del sistema.

2.5 Fecha de Elaboración: Resulta importante el incluir la fecha de elaboración, pues representa un punto de referencia y control.

2.6 Área donde fue elaborado: Incluir el nombre del área en donde fue elaborado el manual.

2.7 Índice del Contenido del Manual: Deberá contar con un índice y/o contenido del manual para facilitar su manejo e identificación de los puntos importantes, pues si sólo se busca un punto en específico con el índice es fácil identificarlo.

2.7.1. Introducción

Breve descripción del sistema desarrollado, que contemple el ámbito abarcado, cual es su función principal y un detalle de las funciones macros o partes que lo componen. Además se debe incluir una pequeña descripción del negocio en el cual se enmarca la aplicación.

2.7.2. Objetivos Generales y Específicos del Sistema

Breve descripción de los objetivos generales y específicos que se cumplieron con el desarrollo del sistema.

2.7.3. Normas, Políticas y Procedimientos

Explicación de las normas, políticas y procedimientos de la organización en las que se basa el sistema para su implementación.

Ejemplo:

El Sistema de Gestión de Recursos Humanos (GREHU) se basa en normas como la NC 3002: 2007. Sistema de Gestión Integrada de Capital Humano-Implementación. El desarrollo del mismo sigue la estrategia establecida por la industria de software cubana de realizar aplicaciones informáticas haciendo uso de las tecnologías de software libre.

2.7.4. Definición de las Reglas del Negocio Implementadas

Definir los lineamientos que se contemplaron durante el desarrollo de la aplicación. Descripción detallada de las reglas de negocio que debe seguir el sistema para garantizar las restricciones que existen en el negocio.

2.7.5. Fundamentación de la Tecnología Utilizada

Realizar una breve descripción, con sus correspondientes referencias bibliográficas, acerca de las tendencias y tecnologías actuales sobre las que se apoya la propuesta, además de incluir la justificación de las seleccionadas para el desarrollo de la aplicación.

2.7.6. Descripción de los Actores del Sistema

Tabla1. Definición Resumida de los Actores del Sistema

Actor del Negocio	Descripción
<Nombre del actor>	Breve descripción del actor con los requerimientos funcionales a él asociados. <i><¿Cómo usa el sistema?. ¿Para qué lo hace?. Identificar características como nivel, la experiencia en la computación, el adiestramiento, o sea, todo lo que resulte significativo.></i>

2.7.7. Especificación de Requisitos

I. Descripción de los Requisitos Funcionales.

Funcionalidades que el sistema debe poseer.

II. Descripción de los Requisitos No Funcionales.

Propiedades o cualidades que el producto debe tener para un mejor rendimiento de la aplicación y satisfacción del usuario. Son las características que hacen al sistema atractivo, usable, rápido y confiable.

Ejemplos:

- *Apariencia o interfaz externa.*
- *Usabilidad.*
- *Rendimiento.*
- *Soporte.*
- *Portabilidad.*
- *Seguridad.*

Si es proporcionada por un sistema externo se debe realizar una revisión bibliográfica del tema y la justificación de su selección. Además se ha de especificar para el sistema la propuesta de: controles de seguridad, privacidad y administrativos, seguridad física y requerimientos funcionales que genera. En caso de que la empresa objeto de estudio cuente con políticas de seguridad bien establecidas y/o sistemas informáticos de seguridad, especificar cómo encaja en ellos el sistema propuesto.

- *Requisitos de implementación.*
- *Requisitos de interface con otros sistemas.*
- *Requisitos de software y hardware.*
- *Restricciones en el Diseño y la Implementación.*
- *Legales.*
- *Confiabilidad.*
- *Ayuda y documentación en línea.*

2.7.8. Vista Funcional

Diagrama de los casos de uso del sistema estructurado que comprometen la arquitectura, incluyendo los de mayor importancia para el usuario. Identificar

relaciones de inclusión, extensión o generalización entre los casos de uso en caso de existir, así como, generalización entre actores si está presente.

Tabla2. Descripción de Bajo Nivel de los Casos de Uso

Nombre del caso de uso	Nombre descriptivo de la funcionalidad que se realiza a través del caso de uso.
Actores	Nombre de los actores, indicando cuál inicia el caso de uso.
Resumen	<p>Descripción del flujo de trabajo. Se debe indicar:</p> <ul style="list-style-type: none"> - ¿Cómo se inicia el proceso? -¿Qué ocurre internamente? -¿Cómo intervienen los actores en el proceso? -¿Cómo termina el proceso? <p>En la descripción han de quedar especificadas las excepciones que se tienen en cuenta y la relación con otros casos de uso, si corresponde.</p>
Precondiciones	Estado inicial. Condiciones que deben ser ciertas antes de ejecutarse el caso de uso. Se refiere además al estado de la información almacenada.
Poscondiciones	Estado final. Condiciones que se generan una vez ejecutado el CU. Se refiere al estado en que queda la información después de ejecutarse el proceso.
Requisitos especiales	Especificar si para el proceso que describe el caso de uso, resulta necesario tener en cuenta restricciones asociadas a la seguridad, velocidad, disponibilidad, exactitud, tiempo de respuesta o utilización de memoria, precisando de que manera lo afecta.

2.7.9 Vista Lógica

* Estilo Arquitectónico: Define la estructuración en capas de la aplicación.

* Patrones de Diseño

Un patrón de diseño es un conjunto de reglas que describen como afrontar tareas y solucionar problemas que surgen durante el desarrollo del software. Existen varias clasificaciones de los patrones, una de ellas es según su finalidad, dentro de la cual se consideran 3 conjuntos fundamentales:

- Patrones de Creación

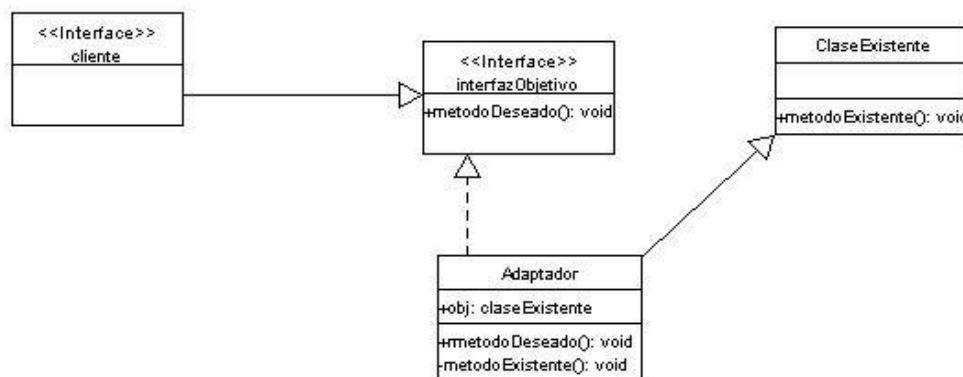
Estos patrones crean objetos, evitando dicha función y la instanciación directa por parte de los desarrolladores. Ello proporciona a nuestros programas una mayor flexibilidad para decidir que objetos usar. Dentro de este grupo los más conocidos son: Factory, Abstract Factory, Builder, Prototype y Singleton.

- Patrones Estructurales

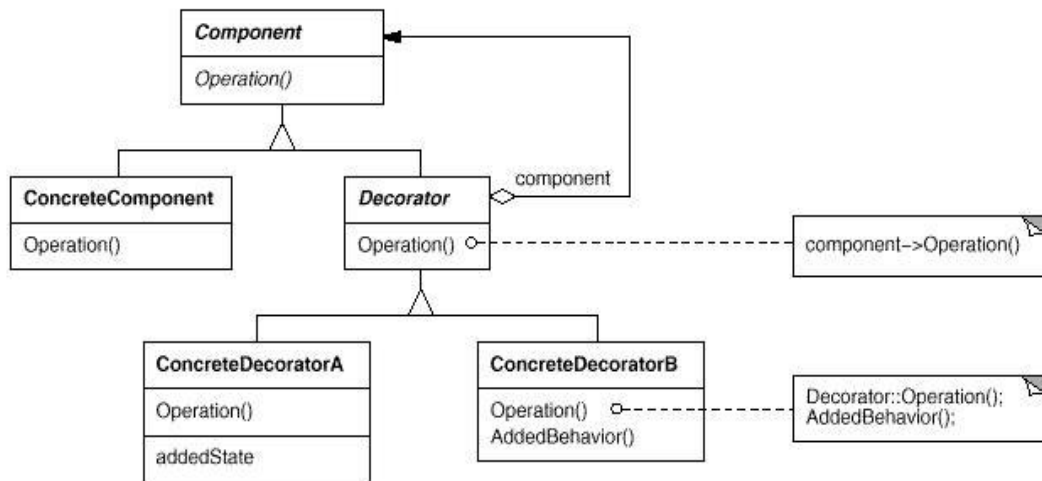
Guían el desarrollo de la aplicación bajo una determinada estructura. Entre ellos se encuentran: Adapter, Bridge, Composite, Decorator, Facade, Flayweight y Proxy.



Estructura del Patrón *Adapter*



✓ Estructura del Patrón *Decorator*



- Patrones de Comportamiento

Enfocados principalmente en algoritmos y en la asignación de responsabilidades entre los objetos. Describen el comportamiento de objetos, clases y la comunicación entre ellos.

Se clasifican

- ✓ **A nivel de clases:** Usan la herencia como vía para distribuir el comportamiento. El Interpreter y el Template Method constituyen ejemplos de este tipo de patrón.
- ✓ **A nivel de objetos:** Usan la composición, más que la herencia para llevar a cabo las tareas. Muestra de ellos son el Command, Iterator, Chain of Responsibility, Strategy y el State.

Patrones de Diseño Detallados

Mencionar los patrones de diseño que se utilizan, especificando para cada patrón:

- a) Breve descripción del patrón.
- b) ¿Dónde y por qué lo usan?

c) Diagrama de clases que muestre cómo se implementa el patrón en la propuesta realizada del sistema.

* *Mecanismos de Diseño*

Son el resultado del refinamiento realizado sobre los mecanismos del análisis. Cabe destacar que dicho proceso se basa en las restricciones impuestas por el ambiente de implementación.

Ejemplos:

- ✓ Mecanismo de Seguridad.
- ✓ Mecanismo de Acceso a Datos.

Mecanismos de Diseño Detallados

- a) Enunciar todos los mecanismos identificados y describirlos brevemente.
- b) Documentar los mecanismos identificados más importantes.
 - I. Vista Estática (Diagrama de clases).
 - II. Descripción de cada elemento mostrado en la Vista Estática.
 - III. Colaboraciones Genéricas con *Diagramas de Secuencia* o de *Colaboración*.

2.7.10. Modelo Lógico de Datos

Incluir el *Diagrama de Clases Persistentes* asociado a la vista de la arquitectura de la aplicación. Dicho diagrama contiene las entidades del sistema, sus atributos, así como las relaciones que se establecen entre ellas. Se puede mostrar por paquetes o casos de uso para ganar claridad en la presentación.

2.7.11. Modelo Físico de Datos

Empleado para describir la estructura de la información persistente manipulada por el sistema. Se debe incluir su respectivo diccionario de datos.

2.7.13. Diseño de Pantallas y Reportes

Breve descripción de las consideraciones asociadas al diseño de las pantallas y reportes de la aplicación.

2.7.14. Descripción de Campos Requeridos por Pantalla

Incluir imágenes de las pantallas.

2.7.15. Vista de Implementación

Compuesta fundamentalmente por un diagrama que muestra las principales piezas desde el punto de vista físico (subsistemas de implementación) que conforman el sistema. Dicho diagrama representa la estructuración física del código, los distintos directorios que organizan el código fuente, librerías, ficheros ejecutables (si existen), entre otros elementos. Se deben describir los componentes mostrados en el diagrama, incluyendo su propósito y contenido. Además han de considerarse como parte de la vista las dependencias con aplicaciones o componentes externos.

2.7.16. Vista de Despliegue

Abarca la representación gráfica del *Diagrama de Despliegue*. Se debe explicar la estructura de los nodos que se presentan e indicar la correspondencia de los mismos con la estructuración en capas implementada.

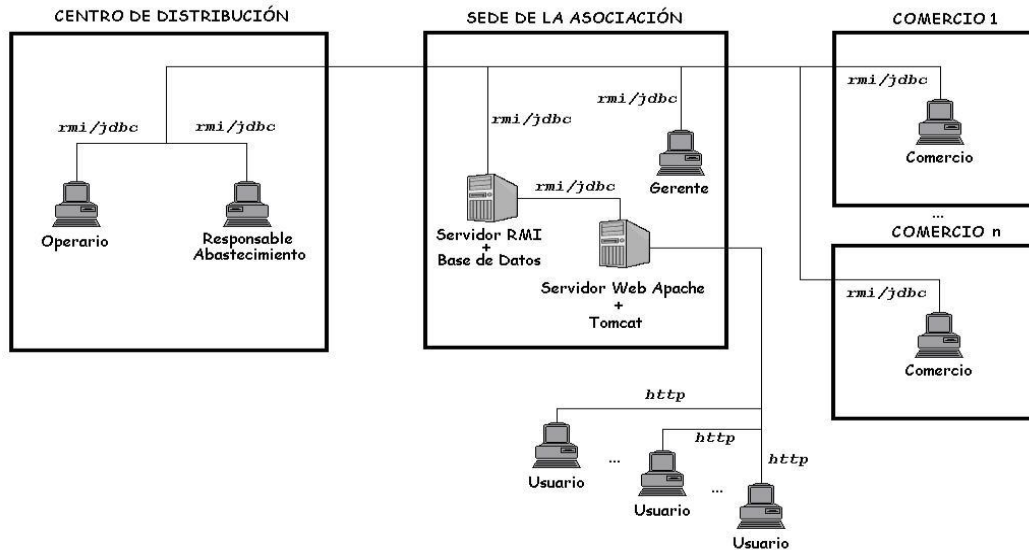


Figura1. Ejemplo de Diagrama de Despliegue

2.7.17. Diagrama de Navegación del Sistema

Muestra gráficamente la interconexión entre cada una de las pantallas del sistema, constituyendo de esta forma los diferentes caminos para llegar a determinada parte de la aplicación. Se deben mostrar los menús, submenús y pantallas a las que nos llevan cada uno de ellos.

2.7.18. Controles de Auditoría Implementados en el Sistema

Adjuntar la documentación vinculada a los controles de auditoría implementados en el sistema.

2.7.19. Glosario de Términos

En él se definen los conceptos y términos importantes para la comprensión del problema, así como de todos los procesos descritos anteriormente.

3. Estándares de Elaboración del Manual

Toda la documentación que se relacione con un sistema, ya sea impresa o digital, sencilla o compleja, debe reunir los siguientes requisitos básicos:

- Debe ser rotulada con claridad y bien organizada en carpetas e índice, con secciones claramente indicadas.
- Los diagramas deberán ser claros, no aglomerados y la escritura manuscrita ha de ser legible.
- La documentación deberá ser completa.
- Se incluirá una leyenda o explicación de los términos utilizados.
- La documentación siempre se conserva actualizada.

El estilo de redacción de los manuales de documentación debe:

- Ser concreto.
- Definir los términos utilizados.
- Utilizar títulos, subtítulos y párrafos cortos.
- Emplear formas activas en lugar de pasivas.
- Aplicar correctamente las referencias bibliográficas.
- No usar frases largas que presenten hechos distintos.