

2

ESTRUCTURA DOCUMENTAL DEL PROYECTO DE INFORMÁTICA.

1.- Introducción.

Una vez concluido el proceso de diseño del proyecto de informática es necesario generar una serie de documentos en los que se exprese la totalidad del proceso seguido en sus dos vertientes: el diseño propiamente dicho y la ejecución de lo diseñado. Es necesario hacer constar que en el gabinete de ingeniería del software se desarrollarán documentos internos durante cada una de las fases del desarrollo del proyecto tendentes a organizar la información existente previamente y la generada en cada fase para facilitar la comunicación entre los miembros integrantes del equipo de diseño y para dejar constancia escrita de aquello que ya está hecho y de lo que queda por hacer, fijando las líneas básicas de actuación para esto último. Sin embargo, en lo sucesivo se tratará del documento resultante de un proyecto ya completado, aunque algunas de sus partes puedan coincidir, al menos en la forma, con los documentos internos antes mencionados. Se trata aquí de dar unas líneas generales para la estructura de este documento y que sirvan de punto de partida para su redacción, aunque cada caso concreto deba recibir un tratamiento individualizado. El objeto principal del documento del proyecto es el de poner de manifiesto el proceso de proyecto seguido, apuntando las principales soluciones y su evaluación hasta llegar a una definitiva para, posteriormente, pasar al desarrollo de la misma y terminando con las estrategias de prueba y validación seguidas.

La estructura del documento será la de un cuerpo principal o "Memoria", en el que se desarrollará lo mencionado anteriormente y unos "Anejos" o "Apéndices" en los que puede aportarse documentación complementaria sobre herramientas usadas o aspectos del proyecto que, por su temática, no tienen cabida en la Memoria, pero que aportan documentación interesante, de tal manera que un lector experto pueda encontrar más información sobre puntos concretos del desarrollo. Asimismo, en los anejos puede darse cuerpo independiente a secciones que así lo requieran, como por ejemplo el código o los manuales de usuario, que normalmente pueden ser utilizados sin haber leído la totalidad de la memoria.

A continuación se desarrolla el contenido pormenorizado que debe tener el documento (figura 5).

MEMORIA	ANEJOS
1.- Introducción	1.- Manual del Usuario
2.- Definición del Problema	2.- Documentación del Código
3.- Objetivos	3.- Documentación de las Pruebas
4.- Antecedentes	4.- Otros.
5.- Restricciones	
- Factores Dato	
- Factores Estratégicos	
6.- Recursos	
- Humanos	
- Hardware	
- Software	
7.- Especificación de Requisitos	
- Datos	
- Interface	
8.- Especificación del Sistema	
- Funciones	
- Costes	
- Agenda	
*.- Conclusiones. En aquellos casos en los que sea necesario	
9.- Bibliografía	

Figura 5. Estructura general del documento.

A.- MEMORIA.

A.1.- INTRODUCCIÓN.

Siempre es conveniente comenzar con una introducción en la que se centren las ideas sobre qué se pretende realizar, aportando información sobre el alcance general de lo desarrollado y su ámbito de aplicación.

Una vez expuesto el problema desde el punto de vista del cliente puede darse una explicación somera que sitúe al lector sobre el tema concreto del proyecto, ya que la temática que puede tocarse en este tipo de proyectos es tan amplia que resulta fácil que escapen a los conocimientos o líneas de trabajo normales de los posibles lectores del documento.

Por ejemplo, si se trata de desarrollar una herramienta de enseñanza asistida por ordenador se comentará el objeto de dichas enseñanzas y la materia de que se trata; si es un entorno de trabajo para procesamiento de imágenes se explicará la utilidad de este tipo de entorno y por qué es necesario desarrollar uno nuevo; etc. Sin embargo no es conveniente entrar en profundidades en este apartado ya que sólo se pretende hacer una “*toma de contacto*” con el tema general del proyecto y posteriormente se hará un desarrollo detallado del mismo.

A.2.- Definición del problema.

En este apartado se entra de lleno en la definición del proyecto desde el punto de vista técnico y se corresponde con la fase de identificación de necesidades y establecimiento de objetivos del proceso general del proyecto. En algunos casos podrá reunirse este apartado con el siguiente (objetivos), aunque esto no debe hacerse de forma general por razones de claridad y lógica interna del documento. Asimismo es necesario separar claramente los problemas real y técnico definiendo ambos de forma clara y concisa.

A.3.- Objetivos.

A continuación se pasa a resumir la información dada anteriormente aportando una relación de aquello que se pretende obtener, dando una descripción de las funciones que debe cumplir lo que se diseña, explicándolas de manera inequívoca y apuntando las posibles vías de solución. En este apartado es necesario ser lo más realista posible definiéndonos a la definición general del problema dada anteriormente con la PDS, y no escribir una larga lista de objetivos ideales pero de difícil consecución. Por otro lado suele darse el caso de que el técnico proyectista espera al final para redactar los objetivos, ya que así está asegurado el éxito con una completa consecución de los mismos, sin embargo este procedimiento no es correcto porque se acometerá el diseño sin haber llegado a definirlo en toda su extensión, lo que acarreará todo tipo de problemas en las fases de prueba y mantenimiento.

A.4.- Antecedentes.

En este apartado se menciona toda aquella información previa y que sirve de base para la ejecución del proyecto, tales como documentación del software existente, datos aportados por la empresa cliente, etc. Debe prestarse especial atención a comentar y criticar constructivamente otras aplicaciones existentes en el mercado sobre el mismo tema, aportando juicios de valor razonados sobre las mismas y mencionando qué se espera del nuevo software que los anteriores no alcancen y cuáles son las causas que motivan su ejecución.

A.5.- Restricciones.

A continuación se expondrán todas aquellas restricciones existentes en el ámbito del diseño y que condicionan la elección de una u otra alternativa. Corresponde a este apartado la identificación de los factores dato y estratégicos mencionados en el capítulo anterior. Los factores dato deben ser aportados por el cliente en cuanto a sus limitaciones presupuestarias y de tiempo, al tipo de hardware existente en su empresa y que no es posible modificar, a los sistemas operativos a emplear y a la definición básica del problema.

Por otro lado, corresponde al técnico proyectista identificar los factores estratégicos que, si bien condicionarán las distintas propuestas alternativas, pueden ser objeto de modificación o elección en uno u otro sentido. Serán factores estratégicos la elección del entorno de trabajo, del lenguaje de programación, herramientas de desarrollo, tipo de interfaz de usuario, etc...

A.6.- Recursos.

En este apartado se expondrá cuáles son los recursos a utilizar para el desarrollo. Se dividirá en recursos humanos (equipo de desarrollo del software con el jefe de proyecto, los analistas de sistemas, comerciales y técnicos, los programadores, etc.) y recursos materiales de hardware y software disponible para la ejecución del proyecto. Es importante, en lo que respecta al equipo del proyecto, clasificar desde el primer momento las distintas tareas a realizar por cada persona para que el trabajo se realice de forma racional sin que existan duplicidades ni lagunas en el trabajo.

A.7.- Especificación de requisitos.

Corresponde a esta parte del documento expresar, desde el punto de vista técnico, las condiciones obtenidas durante el análisis de requisitos, deteniéndose en el *“qué debe hacer el sistema”* y no en el *“cómo debe hacerlo”*, que será el objeto de la Especificación del Sistema. Se responderá aquí a las cuestiones mencionadas en el método PDS aplicado a un proyecto de desarrollo de software. Por otra parte no conviene confundir la especificación de requisitos con el análisis del sistema propiamente dicho, ya que éste representa un conjunto de tareas previo que ha de realizarse, pero que no es el objeto del documento.

A continuación se desglosa el contenido básico de este apartado.

A.7.1.- Descripción general. (Reconocimiento del problema).

En esta información se define el problema especificando cuáles son los objetivos generales del mismo a la vista de toda la información vista anteriormente. No conviene entrar en detalles en esta sección, ya que esto se hará en las inmediatamente posteriores.

A.7.2.- Descripción de la información.

Se realizará aquí una descripción detallada del problema a resolver articulándola en varios apartados o subsecciones: el tipo y flujo de datos a manejar, el flujo de control y la interface de usuario.

En primer lugar se definen los datos, entendidos como la información a manejar y transformar por el programa, especificando si deben ser aportados por el usuario, si serán extraídos de un banco de datos o ambas cosas. A continuación se describe el flujo de control, representando cómo debe responder el sistema a los sucesos o distintos valores que puedan tomar los datos en su transformación, por medio de una representación gráfica y de una explicación literal. Por último nos centraremos en especificar los requisitos del interfaz de usuario, aclarando el tipo de usuarios que utilizarán el sistema y la idea previa que tienen del software, pormenorizando el conjunto de acciones a realizar por los mismos y la manera en que se visualizarán los distintos datos de entrada y salida. Asimismo se aclarará hasta qué punto el interfaz debe ser configurable por el usuario, las ayudas disponibles en cada momento, etc.

Llegado este punto pueden darse ya los requerimientos de hardware necesarios para dar funcionalidad a los requerimientos del sistema.

A.7.3.- Descripción funcional.

Como su nombre indica corresponde a esta sección una descripción de las distintas funciones a implementar para lograr el manejo de datos en la forma deseada. Se describirá cada una por separado expresando a grandes rasgos cuáles son sus entradas y salidas, los posibles sucesos y las acciones requeridas del usuario. Asimismo se indicará cuál debe ser el rendimiento en términos de demanda de recursos del sistema y de los criterios de validación.

A.8.- Especificación del sistema.

La última parte del documento Memoria se centra en la aplicación del diseño a la especificación de requisitos hecha anteriormente. Se detallará en varios apartados el diseño de la interfaz de usuario y el desarrollo final de cada una de las funciones mencionadas en la Especificación de Requisitos, explicando su funcionamiento interno.

Finalmente debe hacerse mención de las pruebas a que se somete el software para comprobar su integridad frente a fallos y la calidad de los resultados obtenidos en el manejo de los datos.

En una subsección puede realizarse un estudio del coste de desarrollo del software y del programa o calendario de tareas necesario para llevarlo a cabo lo más racionalmente posible.

A.9.- Conclusiones.

Este apartado, si bien no obligatorio, es de utilidad para ayudar a los técnicos que tengan que mejorar o mantener el software que se presenta. Su objeto es el de poner de manifiesto los puntos débiles y fuertes del software desarrollado y orientar a los futuros desarrolladores sobre las mejoras a implementar.

A.10.- Bibliografía.

Éste debe ser siempre el último apartado de la Memoria y en él se hará referencia a la documentación utilizada durante la ejecución del proyecto, como por ejemplo libros, revistas, documentación comercial, manuales de programas, páginas web, normativa, etc. La forma de relacionar las referencias será preferentemente por orden alfabético de autores, aunque también puede aparecer numéricamente, en el mismo orden en el que son citadas a lo largo del documento. La norma UNE 50-104-94 regula la forma de citar las referencias a lo largo del texto y se resume a continuación.

Existe una forma general de realizar las referencias bibliográficas en un texto, aunque ésta puede aplicarse de distinta forma según el tipo de documento de que se trate.

Libros: en primer lugar debe referenciarse el autor, identificado por sus apellidos e iniciales del nombre, a continuación aparecerá el título seguido por el traductor y el revisor técnico si los hubiera, el número de edición de que se trata (1ª, 2ª, 3ª, etc.), el lugar de publicación y el nombre del editor o de la empresa editora, el año, la extensión del documento en páginas, la serie, si es que el libro pertenece a una serie de textos relacionados entre sí y publicados por la misma editorial y, por último el número de serie normalizado o ISBN.

Ejemplo: Pressman, R.S. Software Engineering. A Practitioner's Approach. Traducido por Cervigón Ruckaüer, C; Hernández Yáñez, L. Revisión Técnica de Troya Linero, J.M. 3ª ed. Madrid. McGraw-Hill/Interamericana de España S.A. 1994. 824 p. ISBN: 84-481-0026-3.

Partes o secciones de libros con un solo autor: igual que antes aparecerá en primer lugar el autor, seguido por el título de la obra completa, el número de la edición y la identificación del tomo si procede. A continuación el traductor, el revisor, el lugar de publicación y la editorial, el año y, finalmente, la localización de la sección indicando el número del capítulo y las páginas que comprende.

Ejemplo: Puig Torné, J. Proyectos Informáticos. Planificación, Desarrollo y Control. 1ª ed. Madrid. Paraninfo. 1994. Capítulo 3. Cómo se Planifica. pp. 23-57.

Partes de libros con autores independientes: se comienza indicando el autor de la parte citada y su título para, a continuación dar la referencia del libro completo por su autor y su título, finalizando de manera análoga a la mencionada en el tipo anterior.

Artículos en revistas: se dará el nombre del autor y el título del artículo. A continuación el nombre genérico de la revista y la localización del artículo dentro de la misma con el año, volumen y páginas que comprende.

Ejemplo: Mateos, M. Aislamiento Acústico a Ruido Aéreo del Cerramiento Acristalado. Montajes e Instalaciones. Febrero de 1997. Año XXVII. Núm. 303. pp. 91-93.

En aquellos casos en que los autores sean más de tres, aparecerán sólo los primeros, hasta un máximo de tres, seguido por la abreviatura latina "*et al.*" (et alii) o alguna expresión equivalente, entre las que se acepta actualmente "*y col.*" (y colaboradores).

La referencia completa de cada cita que aparece en el texto puede aparecer a pie de página y como lista al final de cada capítulo o al final del texto completo. Asimismo, para confeccionar las listas puede elegirse hacerlo por orden alfabético o por el orden en que aparecen en el texto. Si se elige el orden alfabético puede darse el caso de que varias referencias consecutivas tengan los mismos autores, en este caso puede sustituirse el nombre del o de los autores por un guión a partir de la segunda referencia.

Se entiende por cita a una referencia abreviada inserta en el texto entre paréntesis (Guerrero, 1997) o mediante un número, normalmente en superíndice, que debe corresponderse unívocamente con un elemento de la lista final. Si se toma la primera opción de citar el autor y el año, es posible que sea necesario citar varias referencias en las que coincidan estos dos elementos, se tomará entonces la solución de seguir las referencias con letras minúsculas consecutivas, (Guerrero, 1997a), (Guerrero, 1997b) etc. Si es ne-

cesario que el mismo libro o artículo aparezca más de una vez citado en el texto, siempre se utilizará la misma cita, aunque se rompa el orden numérico si se utilizan números.

B. ANEJOS.

B.1.- Manual de usuario.

La misión del manual de usuario es la de indicar al mismo cómo instalar y manejar el programa de la forma más sencilla y precisa posible. Para su confección se comenzará con la instalación, que se explicará paso a paso, mencionando cuales son los comandos a ejecutar y ayudándose de figuras explicativas de lo que *ha de verse* en pantalla en cada etapa del proceso. A continuación se mostrará cuál es el funcionamiento del programa siguiendo las mismas directrices anteriores: claridad y sencillez. Se explicará la estructura de la interfaz de usuario, la introducción de datos, cuáles son las opciones configurables, la forma de imprimir y salvar los datos, etc. En definitiva el manual de usuario debe ser suficiente como para que cualquier persona no experta pueda desenvolverse sin problemas en el manejo del programa. Por ello, es muy conveniente que el manual se ilustre con ejemplos concretos que aclaren al usuario cómo debe proceder en cada sección.

Como se ha dicho, el manual de usuario debe poder ser utilizado sin necesidad de leer la memoria del proyecto, ya que su finalidad es completamente distinta. Por ello es conveniente que el manual de usuario aparezca como un documento encuadernado en volumen aparte al de la memoria.

B.2.- Documentación del código.

La documentación del código debe ir más allá de un simple listado del código fuente del programa, de tal manera que una persona distinta del autor pueda identificar con facilidad el significado de las variables y el funcionamiento de las funciones (datos de entrada y salida, llamadas a otras funciones, etc.) sin necesidad de realizar un tedioso trabajo de interpretación. Así, es de utilidad asignar a las variables nombres largos y no sólo caracteres individuales, lo que mejora notablemente la comprensión de las mismas. Por otro lado es conveniente hacer comentarios siempre que se estime oportuno al objeto de poder recordar con facilidad el significado de cada parte durante las fases de prueba y mantenimiento.

Como regla general se utilizarán *comentarios de prólogo* o *comentarios de cabecera* al principio de cada módulo y que contengan al menos la siguiente información:

- Título indicativo de la función del módulo.

- Ejemplo de llamada.
- Descripción de los argumentos utilizados (entradas y salidas).
- Lista de los módulos subordinados o relacionados.
- Explicación literal del flujo de datos interno del código.
- Historia del desarrollo conteniendo el autor, el revisor y el listado de las modificaciones realizadas con su fecha y una breve descripción de las mismas.

Además de estos comentarios de introducción a cada módulo del programa, es útil insertar *comentarios descriptivos* entre las líneas del código de tal manera que se aclare el significado de las mismas. Por otro lado es necesario asegurarse de que las líneas de comentario son perfectamente legibles, por ejemplo realizando un sangrado de las mismas, y de que aclaran suficientemente el contenido del programa.

Otro aspecto a tener en cuenta a la hora de escribir el código es el del aspecto general que se da a las líneas del programa mediante sangrados, líneas en blanco y cualquier otra estrategia que añada variantes visuales a la codificación para su mejor lectura e interpretación. Sin embargo no existen reglas fijas sobre cómo realizar esto, por lo que lo más aconsejable es utilizar herramientas de generación automática del código que, si bien no realizan todo el trabajo ni mucho menos, ayudan en la generación de las directrices principales del código, como por ejemplo los comentarios de cabecera y el sangrado de líneas.

B.3.- Documentación de las pruebas.

Se trata aquí de dar una idea precisa de las estrategias de prueba seguidas en la verificación del software y de los resultados obtenidos en las mismas. El proceso de las pruebas consiste en someter al software a una evaluación para comprobar que se comporta de acuerdo a las especificaciones realizadas antes de la fase de diseño. La documentación de las pruebas constará de una descripción en la que se especifica el objeto de la prueba, sobre qué parte del software se desarrolla y los resultados esperados, así como un listado de los resultados obtenidos en las pruebas de cada parte (interfaz de usuario, módulos, comunicaciones con los periféricos, etc.) y de las modificaciones realizadas en virtud de los mismos. Una vez dado por bueno el software, se apuntarán las conclusiones generales obtenidas sobre el desarrollo realizado, comentando el grado de consecución de los objetivos iniciales propuestos y dando las líneas generales de mejora para futuros desarrollos.

B.4.- Otros anexos.

A lo largo del desarrollo del software, y teniendo en cuenta la complejidad y diversidad de los temas que pueden ser objeto de estudio en este campo, es posible que aparezcan

aspectos que no responden directamente al desarrollo propiamente dicho, pero que pueden ser de utilidad a lectores expertos en la materia de que se trata. Esta información adicional se desarrollará en cuantos documentos anexos sea necesario teniendo siempre en cuenta que debe especificarse información que ha sido estudiada o desarrollada por el equipo de desarrollo del software, y no desarrollar conocimientos o materias de ámbito general que puedan encontrarse en los libros especializados en cada tema.