

TEMA 1 Documentación de programas.

1 LA DOCUMENTACIÓN DE PROGRAMAS

En la ejecución de un proyecto informático o un programa software se deben de seguir una serie de pasos desde que se plantea el problema hasta que se dispone del programa o del a aplicación funcionando en el ordenador.

Los pasos son los siguientes:

- Análisis de factibilidad
- Análisis de requerimientos
- Diseño del sistema
- Implementación
- Validación y pruebas
- Explotación
- Mantenimiento

Cada uno de estos pasos debe de llevar asociado un documento. Estos documentos son muy importantes ya que van a regir las fases del ciclo de vida del software y se recogen los pasos seguidos en cada fase para su ejecución.

No es viable la solución mostrada por algunos programadores de ir directamente a la implementación sin antes pararse en la fases 1, 2 y 3. Un trabajo deficiente en estas fases supone una mala definición del problema y por tanto el sistema no cumplirá seguramente con todos los requisitos. El diseño del sistema no será efectivo y los errores serán de difícil solución.

Por lo tanto en la realización de las prácticas será obligado cumplimentar un formulario que guiará al alumno en la fase de análisis de requisitos y de diseño.

2 EL DOCUMENTO DE ESPECIFICACIONES

Este documento tiene como objeto asegurar que tanto el desarrollador como el cliente tienen la misma idea sobre las funcionalidades del sistema. Es muy importante que esto quede claro ya que si no el desarrollo software no será aceptable.

En el caso de este curso, es importante que el alumno y el profesor tengan la misma idea de que hay que desarrollar en la práctica, si un alumno no desarrolla lo que el profesor espera no obtendrá una nota adecuada con su expectativa. Por lo tanto es muy importante que las especificaciones del problema estén claras por ambos.

Existe una normativa referente a este tipo de documento, en Ingeniería del Software I se os dará con más detalle, aquí sólo se intenta que se entienda el alcance e importancia de este documento.

Según la norma IEEE 830, un ERS debe contener los siguientes puntos:

- I. Introducción (Se definen los fines y los objetivos del software)
 - A. Referencia del sistema
 - B. Descripción general
 - C. Restricciones del proyecto
- II. Descripción de la **información** (Descripción detallada del problema, incluyendo el HW y SW necesario)
 - A. Representación del flujo de la información.
 1. Flujo de datos
 2. Flujo de control
 - B. Representación del contenido de la información.
 - C. Descripción de la interfaz del sistema.
- III. Descripción **funcional** (Descripción de cada función requerida, incluyendo diagramas)
 - A. Partición funcional
 - B. Descripción funcional
 1. Narrativa de procesamiento
 2. Restricciones/Limitaciones.
 3. Requisitos de rendimiento.
 4. Restricciones de diseño
 5. Diagramas de soporte
 - C. Descripción del control
 1. Especificación del control
 2. Restricciones de diseño
- IV. Descripción del **comportamiento** (comportamiento del SW ante sucesos externos y controles internos)
 - A. Estados del sistema
 - B. Sucesos y acciones

- V. Criterios de validación.
 - A. Límites de rendimiento
 - B. Clases de pruebas
 - C. Respuesta esperada del SW
 - D. Consideraciones especiales
- VI. Bibliografía
- VII. Apéndice.

3 EL DOCUMENTO DE DISEÑO

En la fase de diseño se toman aquellas decisiones relativas a la futura implementación, se decide la estructura de datos a utilizar, la forma en que se van a implementar las distintas estructuras, el contenido de las clases (sus métodos, los atributos, ...), los objetos. También se definen las funciones, sus datos de entrada y salida, que tarea realizan, para alguna de especial interés el algoritmo que soluciona el problema.

El flujo del programa se define mediante una serie de gráficos que permiten visualizar cual es la evolución del sistema software, en caso de orientación de objetos existen el diagrama de clases, el diagrama de importante tenerlo claro para ello existen una serie de diagramas que permitan clarificar este asunto.

4 LA DOCUMENTACIÓN DEL CÓDIGO FUENTE

Durante la fase de implementación, cuando se está programando, es necesario comentar convenientemente cada una de las partes que tiene el programa. Estos comentarios se incluyen en el código fuente con el objeto de clarificar y explicar cada elemento del programa, se deben de comentar las clases, las variables, los módulos y en definitiva todo elemento que se considere importante.

Esta documentación tiene como objeto hacer más comprensible el código fuente a otros programadores que tengan que trabajar con él, ya sea porque forman parte del grupo de desarrollo, el programa va a ser mantenido o modificado por otra persona distinta al programador inicial. También resulta muy útil durante la depuración y el mantenimiento del programa por el propio programador, al paso del tiempo las decisiones se olvidan y surgen dudas hasta en el propio programador de porqué se hicieron las cosas de una determinada manera y no de otra.

5 FORMULARIO DE PRÁCTICAS

Antes del comienzo de cada práctica es necesario haber realizado un primer estudio del problema a resolver durante la sesión, para ello es obligatorio el cumplimentar este formulario. El objetivo del documento es asegurar que el alumno ha analizado la práctica y ha madurado suficientemente el problema como para estar capacitado para afrontar la codificación del programa.

El documento a entregar es el siguiente:

Descripción del problema:

| |
|--|
| |
|--|

Restricciones del problema

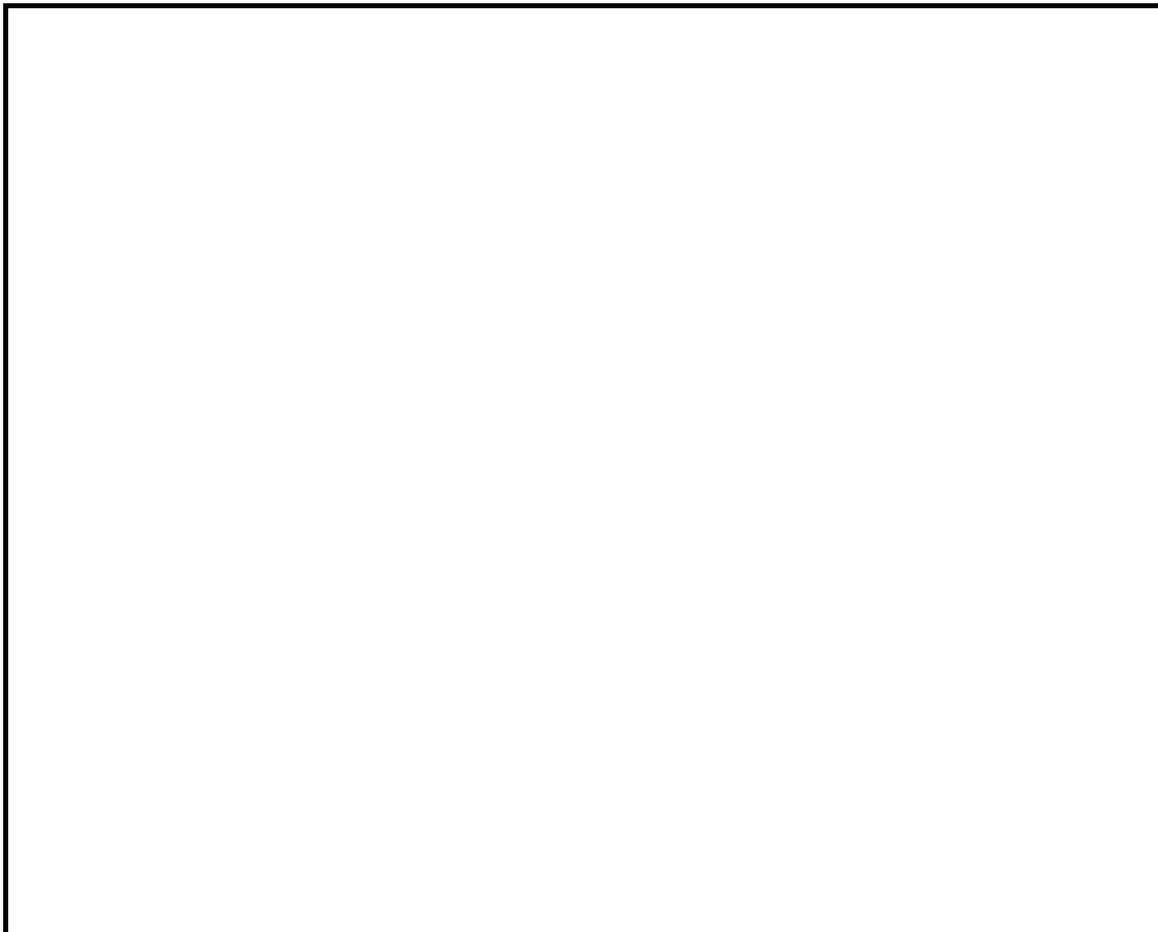
| |
|--|
| |
|--|

Definición de las clases

| | | |
|--|---------------------------------|------------------------------------|
| <i>Nombre de la Clase:</i> | | |
| <i>Métodos Públicos</i> | | |
| <i>Salida (Valor devuelto)</i> | <i>Nombre del método</i> | <i>Entrada (argumentos)</i> |
| <i>Descripción (Responsabilidad):</i> | | |
| <i>Salida (Valor devuelto)</i> | <i>Nombre del método</i> | <i>Entrada (argumentos)</i> |

| | | |
|---|--------------------------|-----------------------------|
| | | |
| <i>Descripción (Responsabilidad):</i> | | |
| <i>Salida (Valor devuelto)</i> | <i>Nombre del método</i> | <i>Entrada (argumentos)</i> |
| <i>Descripción (Responsabilidad):</i> | | |
| <i>Dependencias con otras clases(Colaboración):</i> | | |
| <i>Atributos</i> | | |
| <i>Nombre</i> | <i>Descripción</i> | |
| | | |

Organigrama del main



Funciones

| | |
|---------------------|---------------|
| <i>Descripción:</i> | |
| <i>Prototipo</i> | |
| <i>Entrada</i> | <i>Salida</i> |
| <i>Descripción:</i> | |
| <i>Prototipo</i> | |
| <i>Entrada</i> | <i>Salida</i> |
| <i>Descripción:</i> | |
| <i>Prototipo</i> | |
| <i>Entrada</i> | <i>Salida</i> |