

Entornos de desarrollo

Fases de desarrollo

MARTA GARRIDO VEGA
marta.garrido@iessanandres.com
IES SAN ANDRÉS
CURSO 2019-2020

Fases

- **ANÁLISIS DE REQUISITOS.**

Se especifican los requisitos funcionales y no funcionales del sistema.

- **DISEÑO.**

Se divide el sistema en partes y se determina la función de cada una.

- **CODIFICACIÓN.**

Se elige un Lenguajes de Programación y se codifican los programas.

- **PRUEBAS.**

Se prueban los programas para detectar errores y se depuran.

- **DOCUMENTACIÓN.**

De todas las etapas, se documenta y guarda toda la información.

Fases

- **EXPLOTACIÓN.**

Instalamos, configuramos y probamos la aplicación en los equipos del cliente.

- **MANTENIMIENTO.**

Se mantiene el contacto con el cliente para actualizar y modificar la aplicación el futuro.

Análisis

- Esta es la primera fase del proyecto. Una vez finalizada, pasamos a la siguiente (diseño).
- Es la fase de mayor importancia en el desarrollo del proyecto y todo lo demás dependerá de lo bien detallada que esté. También es la más complicada, ya que no está automatizada y depende en gran medida del analista que la realice.
- **Es la primera etapa del proyecto, la más complicada y la que más depende de la capacidad del analista.**
- ¿Qué se hace en esta fase?

Se especifican y analizan los requisitos funcionales y no funcionales del sistema.
- **Requisitos:**
 - **Funcionales:** Qué funciones tendrá que realizar la aplicación. Qué respuesta dará la aplicación ante todas las entradas. Cómo se comportará la aplicación en situaciones inesperadas.
 - **No funcionales:** Tiempos de respuesta del programa, legislación aplicable, tratamiento ante la simultaneidad de peticiones, etc.

La culminación de esta fase es el documento **ERS** (Especificación de Requisitos Software).

Análisis ejemplo

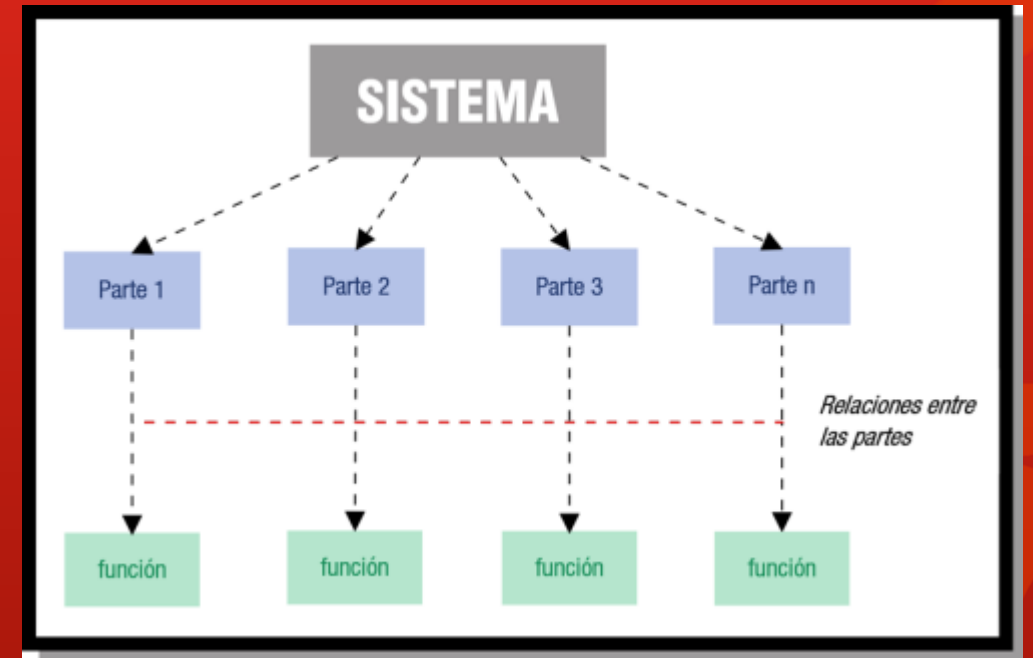
Como ejemplo de requisitos funcionales, en la aplicación para nuestros clientes de las tiendas de cosmética, habría que considerar:

- Si desean que la lectura de los productos se realice mediante códigos de barras.
- Si van a detallar las facturas de compra y de qué manera la desean.
- Si los trabajadores de las tiendas trabajan a comisión, tener información de las ventas de cada uno.
- Si van a operar con tarjetas de crédito.
- Si desean un control del stock en almacén.
- Etc.

Diseño

Durante esta fase, donde ya sabemos lo que hay que hacer, el siguiente paso es ¿Cómo hacerlo?

- Se debe dividir el sistema en partes y establecer qué relaciones habrá entre ellas.
- Decidir qué hará exactamente cada parte.
- En definitiva, debemos crear un modelo funcional-estructural de los requerimientos del sistema global, para poder dividirlo y afrontar las partes por separado.
- Selección del lenguaje de programación que se va a utilizar
- Selección del Sistema Gestor de Base de Datos.
- Etc.



Codificación

- Durante la fase de codificación se realiza el proceso de programación. Consiste en elegir un determinado lenguaje de programación, codificar toda la información anterior y llevarlo a código fuente.
- **Esta tarea la realiza el programador y tiene que cumplir exhaustivamente con todos los datos impuestos en el análisis y en el diseño de la aplicación.**

Las **características** deseables de todo código son:

- Modularidad: que esté dividido en trozos más pequeños.
- Corrección: que haga lo que se le pide realmente.
- Fácil de leer: para facilitar su desarrollo y mantenimiento futuro.
- Eficiencia: que haga un buen uso de los recursos.
- Portabilidad: que se pueda implementar en cualquier equipo.

Estados del código

- **Código Fuente:** es el escrito por los programadores en algún editor de texto. Se escribe usando algún lenguaje de programación de alto nivel y contiene el conjunto de instrucciones necesarias.
- **Código Objeto:** es el código binario resultado de compilar el código fuente. La compilación es la traducción de una sola vez del programa, y se realiza utilizando un compilador. La interpretación es la traducción y ejecución simultánea del programa línea a línea. El código objeto no es directamente inteligible por el ser humano, pero tampoco por la computadora. Es un código intermedio entre el código fuente y el ejecutable y sólo existe si el programa se compila, ya que si se interpreta (traducción línea a línea del código) se traduce y se ejecuta en un solo paso.
- **Código Ejecutable:** Es el código binario resultante de enlazar los archivos de código objeto con ciertas rutinas y bibliotecas necesarias. El sistema operativo será el encargado de cargar el código ejecutable en memoria RAM y proceder a ejecutarlo. También es conocido como código máquina y ya sí es directamente inteligible por la computadora.

Los programas interpretados no producen código objeto. El paso de fuente a ejecutable es directo.

Fases en la obtención de código: **Fuente**, Objeto y Ejecutable

El código **fuentes** es el conjunto de instrucciones que la computadora deberá realizar, escritas por los programadores en algún lenguaje de alto nivel.

- Este conjunto de instrucciones no es directamente ejecutable por la máquina, sino que deberá ser traducido al lenguaje máquina, que la computadora será capaz de entender y ejecutar.
- Un aspecto muy importante en esta fase es la elaboración previa de un algoritmo, que lo definimos como un conjunto de pasos a seguir para obtener la solución del problema. El algoritmo lo diseñamos en pseudocódigo y con él, la codificación posterior a algún Lenguaje de Programación determinado será más rápida y directa.

Fases en la obtención de código: **Fuente**, Objeto y Ejecutable

Para obtener el código fuente de una aplicación informática:

- Se debe partir de las etapas anteriores de análisis y diseño.
- Se diseñará un algoritmo que simbolice los pasos a seguir para la resolución del problema.
- Se elegirá una Lenguajes de Programación de alto nivel apropiado para las características del software que se quiere codificar.
- Se procederá a la codificación del algoritmo antes diseñado.

Fases en la obtención de código:

Fuente, Objeto y Ejecutable

- Puesto que, como hemos dicho antes, este código no es inteligible por la máquina, habrá que TRADUCIRLO, obteniendo así un código equivalente pero ya traducido a código binario que se llama código objeto. Que no será directamente ejecutable por la computadora si éste ha sido compilado.
- Un aspecto importante a tener en cuenta es su licencia. Así, en base a ella, podemos distinguir dos tipos de código fuente:
 - Código fuente abierto. Es aquel que está disponible para que cualquier usuario pueda estudiarlo, modificarlo o reutilizarlo.
 - Código fuente cerrado. Es aquel que no tenemos permiso para editarlo.

Fases en la obtención de código: Fuente, **Objeto** y Ejecutable

- El código objeto es un código intermedio.
- Es el resultado de traducir código fuente a un código equivalente formado por unos y ceros que aún no puede ser ejecutado directamente por la computadora. Es decir, es el código resultante de la compilación del código fuente.
- Consiste en un bytecode (*Código binario resultante de la traducción de código de alto nivel que aún no puede ser ejecutado.*) que está distribuido en varios archivos, cada uno de los cuales corresponde a cada programa fuente compilado.
- Sólo se genera código objeto una vez que el código fuente está libre de errores sintácticos y semánticos

Fases en la obtención de código: Fuente, **Objeto** y Ejecutable

- El proceso de traducción de código fuente a código objeto puede realizarse de dos formas:
 - a. **Compilación:** El proceso de traducción se realiza sobre todo el código fuente, en un solo paso. Se crea código objeto que habrá que enlazar. El software responsable se llama compilador
 - b. **Interpretación:** El proceso de traducción del código fuente se realiza línea a línea y se ejecuta simultáneamente. No existe código objeto intermedio. El software responsable se llama intérprete. El proceso de traducción es más lento que en el caso de la compilación, pero es recomendable cuando el programador es inexperto, ya que da la detección de errores es más detallada.

El código objeto es código binario, pero no puede ser ejecutado por la computadora

Fases en la obtención de código: Fuente, Objeto y Ejecutable

- El código ejecutable, resultado de enlazar los archivos de código objeto, consta de un único archivo que puede ser directamente ejecutado por la computadora. No necesita ninguna aplicación externa. Este archivo es ejecutado y controlado por el sistema operativo.
- Para obtener un sólo archivo ejecutable, habrá que enlazar todos los archivos de código objeto, a través de un software llamado linker (*Enlazador. Pequeño software encargado de unir archivos para generar un programa ejecutable.*) y obtener así un único archivo que ya sí es ejecutable directamente por la computadora.

