

# ENTORNOS DE DESARROLLO

**IES Santiago Hernández**  
**Curso 2017-2018**  
**Ignacio Agudo Sancho**

# UD 1. Desarrollo de Software

- ⦿ Introducción
- ⦿ Tipos de software
- ⦿ Ciclo de vida del software (breve)
- ⦿ Fases del desarrollo (breve)
- ⦿ Ejemplos DFD y Pseudocódigo

# UD 1. Desarrollo de Software

- Introducción
- Tipos de software
- Ciclo de vida del software (breve)
- Fases del desarrollo (breve)
- Ejemplos DFD y Pseudocódigo

# Introducción

- ⦿ Ordenador = Hardware + Software
- ⦿ Hardware = tangible
- ⦿ Software = Programas y aplicaciones que actúan sobre el hardware del ordenador y facilitan al usuario la realización de diferentes tareas.

# UD 1. Desarrollo de Software

- Introducción
- Tipos de software
- Ciclo de vida del software (breve)
- Fases del desarrollo (breve)
- Ejemplos DFD y Pseudocódigo

# Tipos de Software

- ⦿ ¿Qué es el software?
- ⦿ Tipos de software
- ⦿ Licencias. Libre y propietario.

# ¿Qué es el Software?

- RAE: “El conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.”
- Estándar 729 IEEE: “El conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

# Tipos de software

- ⦿ Se puede dividir en dos categorías:
  - Según el TIPO DE TAREA QUE REALIZA
    - De sistema
    - De aplicación
    - De programación o desarrollo
  - Según el MÉTODO DE DISTRIBUCIÓN
    - Propietario o privativo
    - Shareware
    - Freeware
    - Adware
    - Software libre
    - De uso específico



# Según el tipo de tarea (1/3)

## ⦿ SW de Sistema:

- Permite que el hardware funcione.
- Administración de la parte física o recursos.
- Usuario<->Componentes
- Ej: S.O., controladores, diagnóstico, corrección...

# Según el tipo de tarea (2/3)

## ⦿ SW de Aplicación

- Para realizar tareas en cualquier campo que pueda ser automatizado o asistido.
- Hace que el ordenador sea una herramienta útil para el usuario.
- Ej: Aplicaciones de control y automatización industrial, aplicaciones ofimáticas, sw educativo, médico, contabilidad, diseño, etc.

# Según el tipo de tarea (3/3)

## ⦿ SW de Programación o Desarrollo

- Proporcionan al programador herramientas para ayudarlo a escribir programas informáticos y a usar diferentes lenguajes de programación de forma práctica.
- Son los llamados IDE (Entornos de Desarrollo Integrados), que agrupan las herramientas en un entorno visual para no tener que hacer todo por comandos (compilar, interpretar, depurar, etc.)

# Según el método de distribución (1/5)

## ⦿ Propietario o PRIVATIVO

- Controlado privadamente.
- No se puede acceder a su código fuente de forma libre.
- El código se encuentra en poder del desarrollador o compañía.
- No se permite su modificación, adaptación ni lectura por parte de terceros.

# Según el método de distribución (2/5)

## ◎ Shareware

- Evaluación gratuita del producto.
- Límite de tiempo de evaluación.
- Restricciones en las capacidades finales.
- Requiere un pago para uso completo.
- También “shareware de precio cero”
  - WinRAR, PC File, Paint Shop Pro, PC-Tools, Virus Scan...

# Según el método de distribución (3/5)

## ⦿ Freeware

- Se distribuye sin coste.
- No confundir con código libre:
  - Mantiene el Copyright.
  - NO se puede modificar.
  - NO se puede utilizar libremente (modificar ni vender).
- Ofrece funcionalidad completa de manera gratuita.
- Suele incluir licencia comercial

# Según el método de distribución (4/5)

## ⦿ Adware

- Software de publicidad.
- Incluidos en programas Shareware.
- Muchas veces se puede evitar su descarga si estamos atentos.
- Comprando la licencia del Shareware también eliminamos la publicidad.

# Según el método de distribución (5/5)

## ⦿ Software Libre

- Puede ser:
  - Utilizado
  - Estudiado
  - Distribuido
  - Mejorado
- No tiene por qué ser gratuito.
- Puede ser distribuido comercialmente.

## ⦿ Muy común utilizar licencia GNU GPL.



# Según el método de distribución (X/5)

## ● Software de Uso Específico

- Se desarrolla para resolver una necesidad determinada en una organización o persona.
- Requiere expertos informáticos para
  - Desarrollo
  - Adaptación
- Ej: Videoclub, Generación de Horarios, Faltas de asistencia, Banca.

## ● Está como X/5 porque realmente no se considera como distribución.

# UD 1. Desarrollo de Software

- ⦿ Introducción
- ⦿ Tipos de software
- ⦿ Ciclo de vida del software (breve)
- ⦿ Fases del desarrollo (breve)
- ⦿ Ejemplos DFD y Pseudocódigo

# Ciclo de vida del software

## ⦿ Existen varias etapas:

- Análisis: Requisitos. ¿Qué queremos?
- Diseño: ¿Cómo lo queremos?
- Codificación: Traducción a código.
- Pruebas: En base a los criterios. Calidad.
- Mantenimiento: Cambios, adaptación, mejoras.

## ⦿ Varios modelos de ciclos de vida (UD 8).

# UD 1. Desarrollo de Software

- ⦿ Introducción
- ⦿ Tipos de software
- ⦿ Ciclo de vida del software (breve)
- ⦿ Fases del desarrollo (breve)
- ⦿ Ejemplos DFD y Pseudocódigo

# Fases del desarrollo

## ⦿ Las fases del desarrollo son:

- Análisis: Requisitos. ¿Qué queremos?
- Diseño: ¿Cómo lo queremos?
- Codificación: Traducción a código.
- Pruebas: En base a los criterios. Calidad.
- Documentación: Reunir docs y clasificar.
- Explotación: Entrega y utilización.
- Mantenimiento: Cambios, adaptación, mejoras.

# UD 1. Desarrollo de Software

- ⦿ Introducción
- ⦿ Tipos de software
- ⦿ Ciclo de vida del software (breve)
- ⦿ Fases del desarrollo (breve)
- ⦿ Ejemplos DFD y Pseudocódigo

# DFD y Pseudocódigo

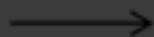
## ◉ Símbolos de Diagrama de Flujo de Datos:



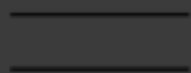
Proceso



Entidad



Flujo de datos



Almacén de datos

# DFD y Pseudocódigo

## ⦿ Reglas de los DFD:

- Los elementos tienen que tener nombres con significado. En el caso de los flujos de datos que entran y salen de un almacén no hace falta indicar nombre.
- Los flujos de datos deben mostrar en qué sentido se mueven los datos.
- Se permiten flujos de datos entre: dos procesos, un proceso y un almacén, un proceso y una entidad externa.



# DFD y Pseudocódigo

## ⦿ Reglas de los DFD:

- No se permiten flujos de datos entre: dos almacenes, dos entidades externas, un almacén y una entidad externa.
- Realizar un DFD para cada parte del sistema que haya sido identificada.
- Los almacenes de datos y entidades externas se pueden representar varias veces en el DFD si con ello mejoran la legibilidad.
- En un DFD no puede haber elementos aislados.

# DFD y Pseudocódigo

- ⦿ Sistema grande = Varios DFD
- ⦿ Estructura jerárquica: Nivel 0.
  - Nivel más alto = único proceso. Nivel 0. Representa el sistema completo.
  - Se presenta el proceso, los flujos de entrada y salida y las entidades externas. No los almacenes.
  - Se llama Diagrama de Contexto o DFD de nivel 0.

# DFD y Pseudocódigo

- ⊙ Estructura jerárquica del DFD: Nivel 1
  - Se descompone el proceso identificado con un 0 en otro DFD en el que se representan las funciones principales del sistema: Nivel 1
  - Se observan ya los flujos de entrada y salida de las entidades externas.
  - No se representan las entidades externas, aunque a veces es conveniente incluirlas para ayudar a la comprensión.

# DFD y Pseudocódigo



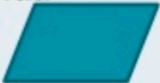





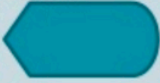
- ⦿ Estructura jerárquica del DFD: Nivel 2
  - Se descompone cada uno de los procesos en nuevos procesos que representan funciones más simples.
  - Se deben mostrar los flujos de entrada y salida del proceso del nivel 1 descompuesto.

# DFD y Pseudocódigo

- ⦿ Se recomienda usar un máximo de cuatro niveles de descomposición:
  - Nivel 0: Diagrama de contexto
  - Nivel 1: Subsistemas
  - Nivel 2: Funciones de cada subsistema
  - Nivel 3: Subfunciones asociadas
  - Nivel 4: Procesos necesarios para el tratamiento de cada subfunción.

# DFD y Pseudocódigo

## ◉ Símbolos de Pseudocódigo:

símbolo	Función	Símbolo	Función
Terminal 	Indicar el inicio y fin del diagrama	Teclado 	Introducir datos manualmente por el teclado
Entrada/salida 	Entrada o salida simple de información	Decisión 	Indica operaciones lógicas o de comparación y tienen dos salidas dependiendo del resultado.
Proceso 	Realizar cualquier operación o calculo con la información	Conectores 	Une dos partes del diagrama a la misma o diferente página
Salida a Impresora 	Salida de informacion a la impresora	Flechas de Flujo 	Indica la direccion del flujo de la información
Salida a Pantalla 	Mostrar información de salida a la pantalla		

# DFD y Pseudocódigo

- Ejemplo 1: Programa que lee dos números y muestra la suma en pantalla
- Ejemplo 2: Programa que lee dos números y muestra el mayor en pantalla, si son iguales deberá mostrar un mensaje indicándolo.
- Ejemplo 3: Programa que lee dos números en un proceso repetitivo. Este proceso terminará cuando los números leídos sean iguales.

# DFD y Pseudocódigo

## ⦿ Ejemplo 4:

- Programa que lee 10 números en un proceso repetitivo y muestra la suma. Es necesario declarar variables para contar los números que se van leyendo y para ir guardando la suma. Se deben inicializar a 0.



# DFD y Pseudocódigo

## ● Ejemplo 5:

- Programa que lee registros de un fichero secuencial. Cada registro contiene información de un alumno: Nombre, Curso, Nota.
- El programa debe mostrar por cada registro leído el Nombre, el Curso y la Nota.
- Al final del proceso de lectura debe mostrar la nota media. Esta se calcula sumando las notas de todos los alumnos y dividiéndola por el número de alumnos que hay.
- Se utiliza la estructura repetitiva Mientras-Hacer-Fin mientras.
- Se declaran tres variables, una para contar alumnos, otra para sumar notas y otra para guardar la nota media.

# DFD y Pseudocódigo

## ◉ Ejemplo 6:

- Programa que lee un número de teclado y muestra a qué día de la semana corresponde.
- Si el número leído es 1, visualizará Lunes.
- Para cualquier valor que no sea válido, mostrará No válido.
- Se utilizará la estructura:
  - Según sea <condición> hacer
    - Caso valor 1: instrucciones
    - Caso valor 2: instrucciones
  - Fin según