

Fundamentos de Programación

+ Objetivos del curso



- Construir programas de pequeño porte que involucren rutinas sencillas de entrada y salida de datos, cálculos y procesamiento básicos
- Comprender los fundamentos de los lenguajes de programación, de las metodologías de desarrollo de software así como de las herramientas para desarrollar software.
- Comprender los fundamentos del paradigma de programación estructurada.

+ Objetivos del curso (2)



- Comprender los fundamentos del paradigma de Programación Orientada a Objetos.
- Aplicar estos conocimientos en lenguaje JavaScript.
- Comprender los fundamentos y el funcionamiento básico de un sistema de control de versiones



Temario



Introducción al desarrollo de software



- Definición de software.
- Conceptos básicos de desarrollo de software: roles, actividades, hitos, entregables, metodologías.
- Lenguajes de programación.
- Código fuente y proceso de compilación.
- Entorno de desarrollo.
- Manejo de versiones y concepto de repositorio.
- Evolución de los lenguajes y paradigmas de programación.
- Lenguajes y tecnologías más utilizadas actualmente.



Introducción a la programación



- Instrucciones.
- Tiempo de diseño y tiempo de ejecución.
- Conceptos básicos de programación.
- Estructuras de control.
- Estructuras de repetición.
- Variables y colecciones.
- Funciones y procedimientos.
- Pseudocódigo.

+ Introducción a JavaScript

- Definiciones básicas del lenguaje.
- Estructuras de control y repetición en JS.
- Construcción de programas sencillos en JS.

+ Algoritmos



- Recorridas sobre colecciones.
- Búsquedas sobre colecciones.
- Otros algoritmos y estructuras de datos más complejos.

+ Programación Orientada a Objetos

- Conceptos básicos de OOP:
 - Clase y objeto
 - Memoria dinámica, punteros e identidad
 - Constructores
 - Herencia
- Conceptos básicos y uso de lenguaje de modelado UML:
 - Introducción a UML
 - Diagramas de Clases en UML

+ Programación Orientada a Objetos en JavaScript

- Proyecto entregable con aplicación de conceptos de programación orientada a objetos en lenguaje JS.

+ Programación Orientada a Objetos (Segunda Parte)

- Conceptos avanzados de OOP:
 - Sobrecarga y redefinición
 - Polimorfismo
 - Interfaces
 - Constructores y destructores y su encadenamiento
- Uso de UML para estos conceptos avanzados.
- Aplicaciones y limitaciones de JS frente a OOP.

+ Desarrollo de programas más complejos

- Proyecto entregable aplicando el uso de programación orientada a objetos, UML y los conceptos de JS vistos en el curso.

+ Desarrollo de programas más complejos

- Proyecto entregable aplicando el uso de programación orientada a objetos, UML y los conceptos de JS vistos en el curso.



Introducción al Desarrollo de Software



Definición de software y hardware



Software:

- Serie ordenada de operaciones necesarias para llevar a cabo un proyecto.
- Cada una de las operaciones que, en un orden determinado, ejecutan ciertas máquinas.

Hardware:

- Partes físicas tangibles de un sistema informático



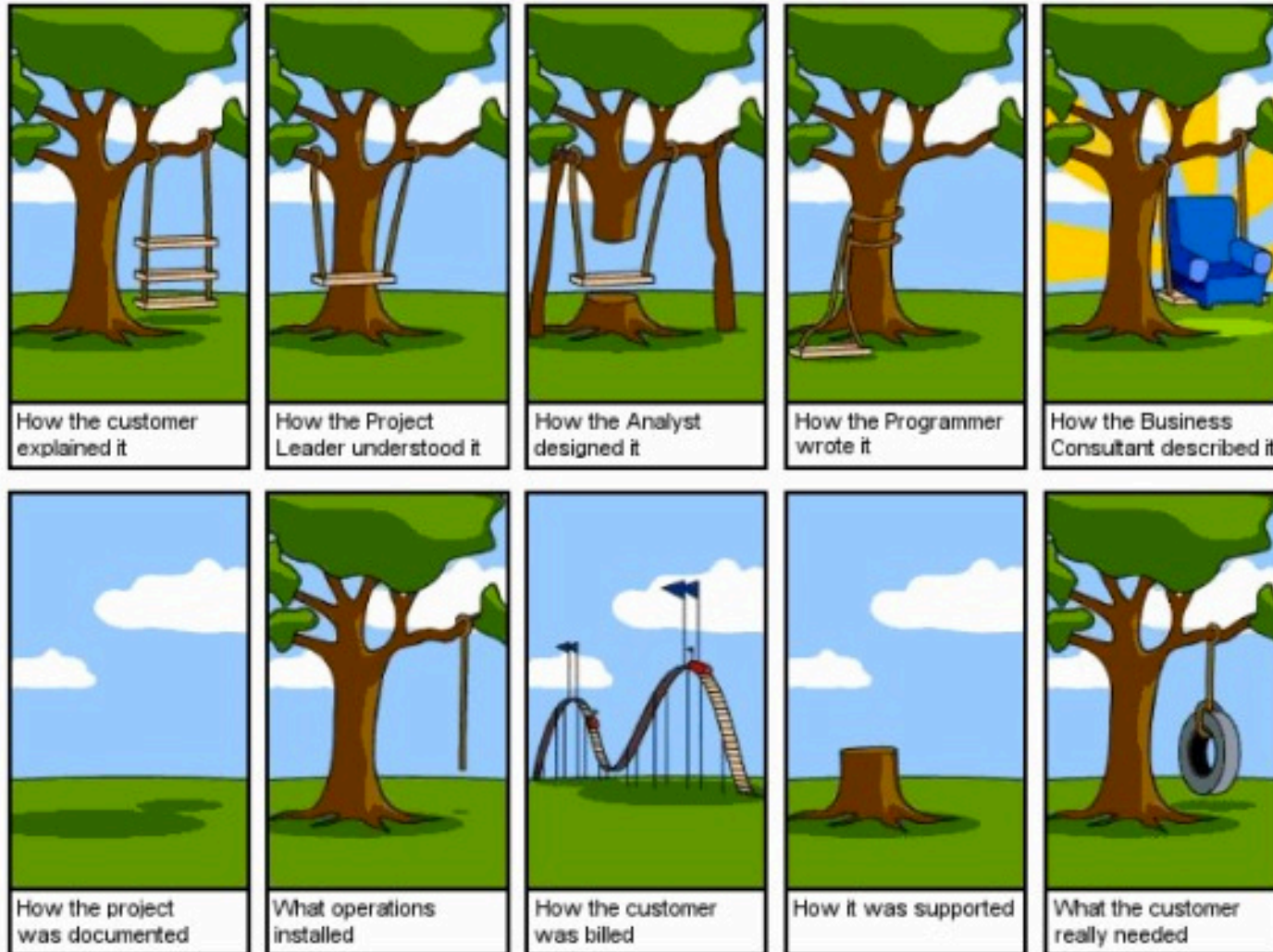
El proceso de desarrollo del software



Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente.

+

El proceso de desarrollo del software (2)





Actividades fundamentales del proceso de desarrollo de software



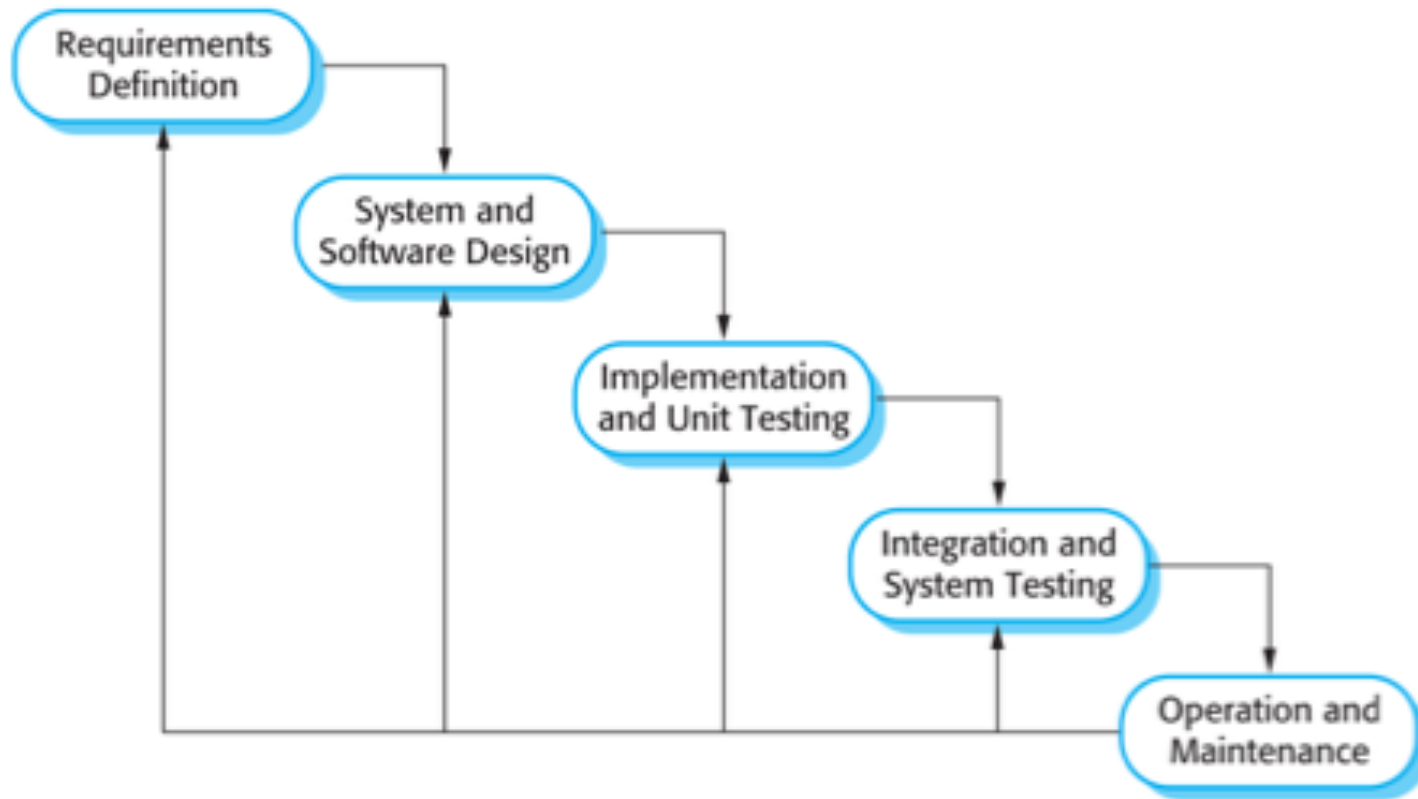
- **Especificación de software:**
Se deben definir las funcionalidades y restricciones que debe cumplir el software a desarrollar.
- **Diseño e Implementación:**
Se diseña y construye el software de acuerdo a la especificación.
- **Validación:**
El software debe validarse, para asegurar que cumpla con lo que quiere el cliente.
- **Evolución:**
El software debe evolucionar, para adaptarse a las necesidades del cliente.

+ Modelos de procesos

Las distintas metodologías se diferencian en términos:

- en base a *QUE* se divide el proyecto
- *CUÁNTAS* etapas tiene
- *QUE* hay que hacer en cada una
- *QUIEN* tiene que hacerlo
- que *INPUT/OUTPUT* tiene cada etapa
- si se *VUELVE ATRÁS* o no .

+ Modelo en cascada



+ Modelo en cascada (2)



- Primer modelo de desarrollo de software
- Fases:
 - Definición de los requisitos
 - Diseño de software
 - Implementación y pruebas unitarias
 - Integración y pruebas del sistema
 - Operación y mantenimiento
- Una fase no comienza hasta que termine la fase anterior



Desventajas del Modelo en cascada



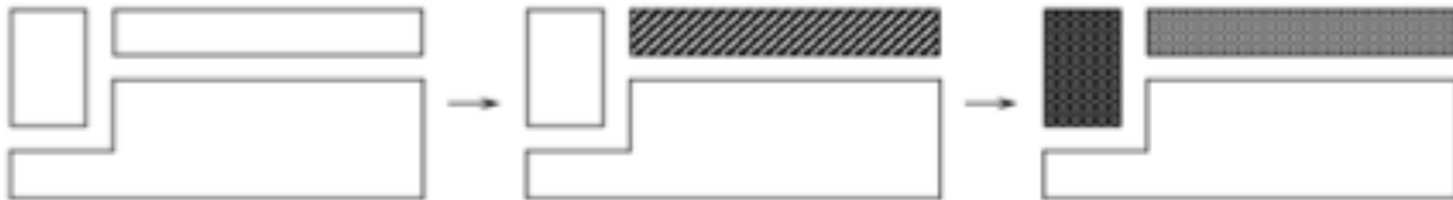
- Nula interacción con el usuario
- Los problemas se dejan para su posterior resolución
- Por lo general el software no cumple con los requisitos del usuario
- Es inflexible a la hora de evolucionar para incorporar nuevos requisitos.

+ Modelo Iterativo e incremental.

DESARROLLO INCREMENTAL



DESARROLLO ITERATIVO





Modelo Iterativo e incremental.



- Particiona el proyecto en Casos de Uso y no en Etapas.
- Reduce la repetición del trabajo en el proceso de desarrollo y retrasa la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema.



Ventajas y Desventajas del modelo Iterativo e Incremental.



- Ventajas :
 - Se libera una versión de forma temprana
 - Los clientes pueden aclarar los requisitos que no tengan claros.
 - Disminuye el riesgo de fracaso de todo el proyecto
 - Las partes más importantes del sistema son entregadas primero, por lo cual se realizan más pruebas en estos módulos y se disminuye el riesgo de fallos.
- Desventajas :
 - Incremento debe ser pequeño para limitar el riesgo
 - Cada incremento debe aumentar la funcionalidad.



Lenguaje de programación



Definición: “Lenguaje formal diseñado para realizar procesos que puedan ser ejecutado por máquinas como las computadoras”

Compilador: “Motor que revisa lo que escribimos en un lenguaje y lo convierte en lenguaje de máquina”.

+ Lenguaje de programación (2)



Debe facilitar alcanzar los objetivos de:

- Corrección
- Claridad
- Eficiencia
- Portabilidad

Categorías :

- de Máquina
- Bajo Nivel
- Alto Nivel

+ ASCII - American Standard Code for Information Interchange

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	á	192	C0	Ł	224	E0	α
129	81	ü	161	A1	í	193	C1	ł	225	E1	β
130	82	é	162	A2	ó	194	C2	ŧ	226	E2	Γ
131	83	â	163	A3	ú	195	C3	ţ	227	E3	π
132	84	ä	164	A4	ñ	196	C4	—	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	+	229	E5	σ
134	86	å	166	A6	ª	198	C6	†	230	E6	μ
135	87	ç	167	A7	º	199	C7	‡	231	E7	ι
136	88	ê	168	A8	¿	200	C8	£	232	E8	φ
137	89	ë	169	A9	ƒ	201	C9	₣	233	E9	Θ
138	8A	è	170	AA	¬	202	CA	₤	234	EA	Ω
139	8B	ï	171	AB	½	203	CB	₦	235	EB	δ
140	8C	î	172	AC	¼	204	CC	₧	236	EC	∞
141	8D	ì	173	AD	¦	205	CD	=	237	ED	φ
142	8E	Ä	174	AE	«	206	CE	÷	238	EE	ε
143	8F	Å	175	AF	»	207	CF	±	239	EF	∩
144	90	É	176	B0	⋮	208	D0	⌚	240	F0	≡
145	91	æ	177	B1	⋮	209	D1	₹	241	F1	±
146	92	Æ	178	B2	⋮	210	D2	₺	242	F2	≥
147	93	ø	179	B3		211	D3	€	243	F3	≤
148	94	ö	180	B4	└	212	D4	Ö	244	F4	[
149	95	õ	181	B5	┐	213	D5	ƒ	245	F5]
150	96	û	182	B6	┌	214	D6	₣	246	F6	÷
151	97	ù	183	B7	└	215	D7	₤	247	F7	≈
152	98	ÿ	184	B8	┐	216	D8	₦	248	F8	≈
153	99	Û	185	B9	┌	217	D9	₧	249	F9	·
154	9A	Ü	186	BA		218	DA	₣	250	FA	·
155	9B	ϕ	187	BB	┐	219	DB	■	251	FB	√
156	9C	£	188	BC	┌	220	DC	■	252	FC	²
157	9D	¥	189	BD	└	221	DD	■	253	FD	²
158	9E	₣	190	BE	┐	222	DE	■	254	FE	■
159	9F	ƒ	191	BF	└	223	DF	■	255	FF	