

## **CICLO DE LA VIDA Y METODOLOGIA DEL SOFTWARE**

**1**

### **CONCEPTO DEL CICLO DE VIDA DEL DESARROLLO DEL SOFTWARE**

También es conocido como SDLC o Systems Development Life Cycle).

Contempla fases necesarias para validar el desarrollo del software para garantizar que este cumpla los requisitos para la aplicación y verificación de los procedimientos de desarrollo, asegurándose de que los métodos usados son apropiados.

**2**

### **FASES DEL DESARROLLO DEL SOFTWARE**

Esta es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con grandes posibilidades de éxito. También indica cómo se divide un proyecto en módulos mas pequeños para normalizar como se administra el mismo.

Una metodología para el desarrollo del software son los procesos a seguir sistemáticamente para idear, implementar y mantener un producto de software desde que surge su necesidad hasta que cumple el objetivo por el cual fue creado.

**3**

### **Planificación**

Es necesario hacer ciertas tareas Dichas tareas son conocidas como el fuzzy front-end del proyecto, puesto que no están sujetas a plazos. También incluye determinación del ámbito del proyecto, la realización de un estudio de viabilidad, el análisis de los riesgos asociados, la estimación del coste del proyecto, su planificación temporal y la asignación de recursos a las diferentes etapas del proyecto.



**4**

### **ANÁLISIS**

Saber qué es exactamente lo que tiene que hacer el software.

Se intenta descubrir qué es lo que realmente se necesita y se llega a una comprensión adecuada de los requerimientos del sistema (las características que el sistema debe poseer).



**5**

### **DISEÑO**

El diseño es una etapa compleja y su proceso debe realizarse de manera iterativa.

Es posible que la solución inicial no sea la más adecuada, por lo que en tal caso hay que refinarla.

Pues siempre hay un diseño iniciativo, pero bien sabemos que la tecnología está muy avanzada así que podemos innovar nuevas ideas.



6

## IMPLEMENTACION

Se eligen herramientas adecuadas, un entorno de desarrollo que facilite el trabajo y un lenguaje de programación apropiado para el tipo de software a construir.

Se deben seguir diferentes pausas como:

- 1.-Evitar bloques de control no estructurados
- 2.-Identificar correctamente las variables y su enlace
- 3.-Eleguir algoritmos y estructuras de datos adecuados para el programa.

7

## PRUEBAS

Detectar los fallos cometidos en las etapas anteriores para corregirlos. Por supuesto, lo ideal es hacerlo antes de que el usuario final se los encuentre. Se dice que una prueba es un éxito si se detecta algún error.



8

## INSTALACIÓN O DESPLIEGUE

En esta fase debemos poner el software en funcionamiento, para eso debemos de planificar el entorno teniendo en cuenta las dependencias existentes entre los diferentes componentes del mismo.



9

## USO Y MANTENIMIENTO

Esta es una de las fases más importantes del ciclo de vida de desarrollo del software. Puesto que el software ni se rompe ni se desgasta con el uso, su mantenimiento incluye los siguientes puntos.

- 1.-Eliminar los defectos detectados durante su vida útil (mantenimiento correctivo).
- 2.-Adaptarlo a nuevas necesidades (mantenimiento adaptativo).
- 3.-Añadirle nuevas funcionalidades (mantenimiento perfecto).

10

## MODELOS DE CICLOS DE VIDA DEL SOFTWARE

Facilitan una metodología común entre el cliente y la compañía de software, los modelos de ciclo de vida se han actualizado para plasmar las etapas de desarrollo involucradas y la documentación necesaria, de forma que cada fase se valide antes de continuar con la siguiente.



11

## MODELO EN CASCADA

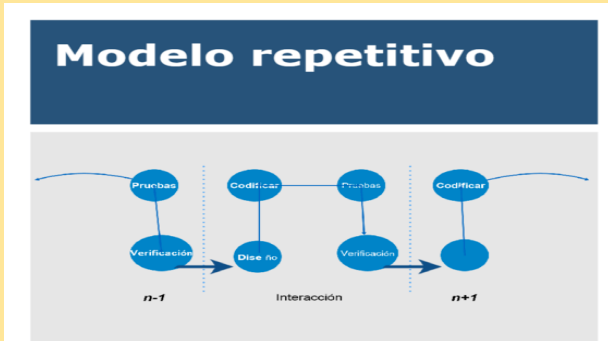
En el modelo de ciclo de vida en cascada las fases anteriores funcionarán una detrás de la otra de manera lineal. De este modo, solo cuando una fase termine se podrá continuar con la siguiente, y así progresivamente.



12

### MODELO REPETITIVO

Este modelo guía el proceso de desarrollo de software en repeticiones. Así, proyecta el proceso de desarrollo de modo cíclico repitiendo cada paso después de cada ciclo en el proceso de ciclo de vida del software.



15

### MODELO BIG BANG

Probablemente este sea el modelo más simple, ya que necesita poca planificación, mucha programación y muchos fondos. Este modelo tiene como concepto principal la creación del universo; así, si se reúnen fondos y programación, se consigue el mejor producto de software.

13

### MODELO EN ESPIRAL

Es una combinación de los modelos anteriores donde se tiene en cuenta el riesgo. De esta forma, se comienza fijando los objetivos y las limitaciones al empezar cada repetición



14

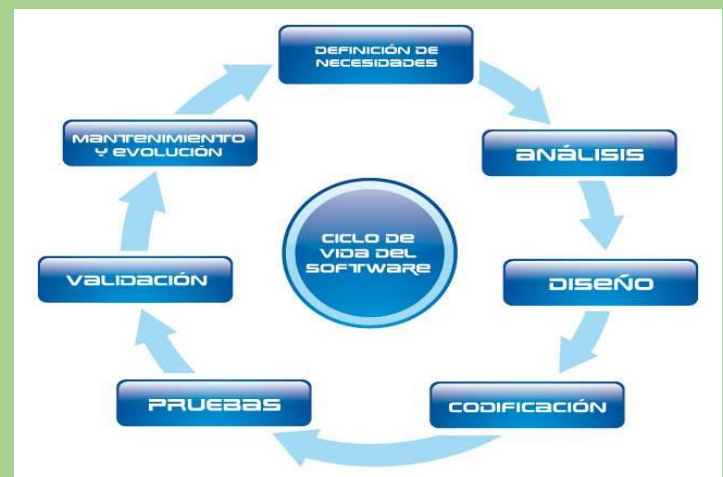
### MODELO EN V

Uno de los grandes problemas del modelo en cascada es que solo se pasa a la siguiente fase si se completa la anterior y no se puede volver atrás si hay errores en etapas posteriores. Así, el modelo en V da más opciones de evaluación del software en cada etapa.



16

### CICLO DE LA VIDA DEL SOFTWARE



17

### REFERENCIAS

ALTVATER, A. L. E. X. A. N. D. R. A. (2020, 8 abril). What Is SDLC. stackify. <https://stackify.com/what-is-sdlc/>

Harnes, H. (2021, 7 octubre). Understanding The Phases Of The Software Development Life Cycle. harness. <https://harness.io/blog/software-development-life-cycle/>

SDLC: Seven Phases of the System Development Life Cycle. (s. f.). Innovativearchitects. Recuperado 9 de octubre de 2021, de <https://www.innovativearchitects.com/KnowledgeCenter/basic-IT-systems/system-development-life-cycle.aspx>

ISO/IEC/IEEE 12207:2017. (2021, 4 febrero). ISO. <https://www.iso.org/standard/63712.html>