**Tarea 2**

**Ciclo de vida del software (Relación 1)**

1.- Define "Ciclo de vida del software".

El ciclo de vida de un software es la secuencia estructurada y bien definida de las etapas del desarrollo del software. Lo describe desde el inicio hasta el final, con el objetivo de definir las distintas fases intermedias para validar el desarrollo de la aplicación y confirmar que cumpla con los requisitos de verificación impuestos por el cliente.



2.- Nombra las fases principales del desarrollo de software y explica brevemente que se hace en cada una de ellas.

PLANIFICACIÓN: En esta fase se trazan los objetivos principales y las necesidades del proyecto. Esta fase principalmente consiste en un estudio de viabilidad, Analisis de errores y planificación de las tareas.

ANÁLISIS: Se determina las funciones que tendrá el software y sus características. También se calcula el coste determinado del desarrollo

DISEÑO: Se estudian las posibles implementaciones que hay que construir y la estructura general del software.

CODIFICACIÓN: Se desarrolla el software previamente desarrollado en código fuente

PRUEBAS: Con el programa inicial ya desarrollado, se utiliza el software enfocado en forzar errores para poder corregirlos

MANTENIMIENTO: Antes de sacar el software, hay que cambiar algunas cosas de este, se puede clasificar de cuatro formas

Correctivo: se corrigen defectos

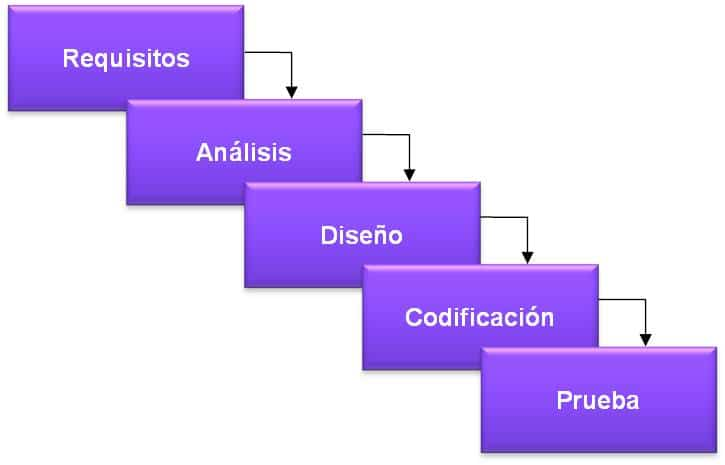
Perfectivo: se mejora la funcionalidad

Evolutivo: se añade funcionalidades nuevas

Adaptativo: se adapta a nuevos entornos.

3.- Explica brevemente en qué consiste el modelo en cascada cuando hablamos de desarrollo de software.

Consiste en aplicar las fases principales del desarrollo una detrás de la otra sin posibilidad de acceder a una fase concreta sin pasar por las anteriores



4.- Ventajas e inconvenientes del modelo en cascada.

Ventajas:

El es una estructura simple y clara

La información se transmite fácilmente entre los ordenadores

El objetivo final se determina rapidamente

Desventajas:

Es inflexible, no deja acceder a las pruebas antes de pasar por todas las fases anteriores

Deja al cliente fuera del desarrollo del software

Dificultar para implementar cambios

5.- ¿Qué se entiende por verificación? ¿Y por validación?

es la parte que garantiza que el resultado del trabajo cumpla con los objetivos propuestos. La verificación del software generalmente incluye:

Conformidad con los estándares de la industria, asegurando que el proceso y los artefactos cumplan con las pautas.

Revisiones, tutoriales, inspecciones.

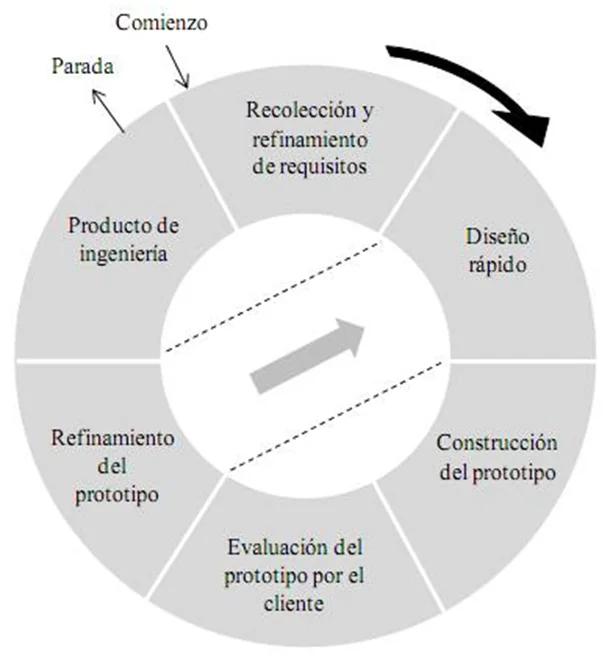
Hacer cumplir los estándares de arquitectura, diseño y codificación

La Validación es la comprobación del software totalmente desarrollado para que cumpla con los estándares de IEEE

6.- Explica cómo funciona el modelo de desarrollo mediante creación de prototipos.

Es un modelo de producción evolutivo que consiste en la generación de programas rápidos de muestra que el cliente podrá ir observando que irán evolucionando hasta llegar a la versión definitiva.

Es un sistema que ayuda al cliente a poder ver mejor como esta yendo el software que encargo.



7.- Explica cómo funciona el modelo espiral cuando se aplica al desarrollo orientado a objetos.

Es un modelo que consiste en la división de las fases en bucles que se van repitiendo constantemente hasta terminar el software, pero en cada bucle es más corto que el anterior porque cada vez queda menos trabajo que terminar. Se llama así porque su representación esquemática se ve como una espiral

