

Tema 2-Ciclos de vida

El software evoluciona a lo largo del tiempo. Los requisitos de gestión, mercado, etc. hacen imposible que se pueda finalizar completamente un producto. Esto conlleva a lanzar versiones limitadas para cubrir la competencia. Los detalles de extensión del producto tienen que pulirse más. Los [Modelos básicos](#) no nos servirían.

Los modelos evolutivos se caracterizan por no entregar un sistema completo. Son iterativos, es decir, permiten desarrollar cada vez versiones más completas del software.

Es importante diferenciar una iteración de un incremento. Una iteración crea una nueva versión del producto. Un incremento añade una nueva funcionalidad.

Modelo incremental

Consiste en crear una implementación parcial del sistema e ir aumentándole funcionalidades. Es una aplicación reiterada de varias secuencias en cascada. Cada ciclo crea un producto operativo, con una nueva funcionalidad o un refinamiento de una anterior función. El primer ciclo aborda los requisitos fundamentales.

Primero se hace el primer incremento. El cliente lo evalúa y elabora un plan para el siguiente incremento. Esto se repite hasta que se elabore el producto completo. Para ello, el software debe construirse de forma que facilite la incorporación de nuevos requisitos.

Ventajas:

- No es necesario tanto personal como para cubrir una implementación completa.
- Permite reducir el gasto antes de conseguir que el producto tenga cierta capacidad.
- Reduce la posibilidad de que los requisitos del cliente cambien tanto.
- Flexibilidad.
- La calidad del software debe mejorar con cada incremento.
- Ayuda en proyectos que utilicen tecnologías a las que la organización no está tan adaptada.

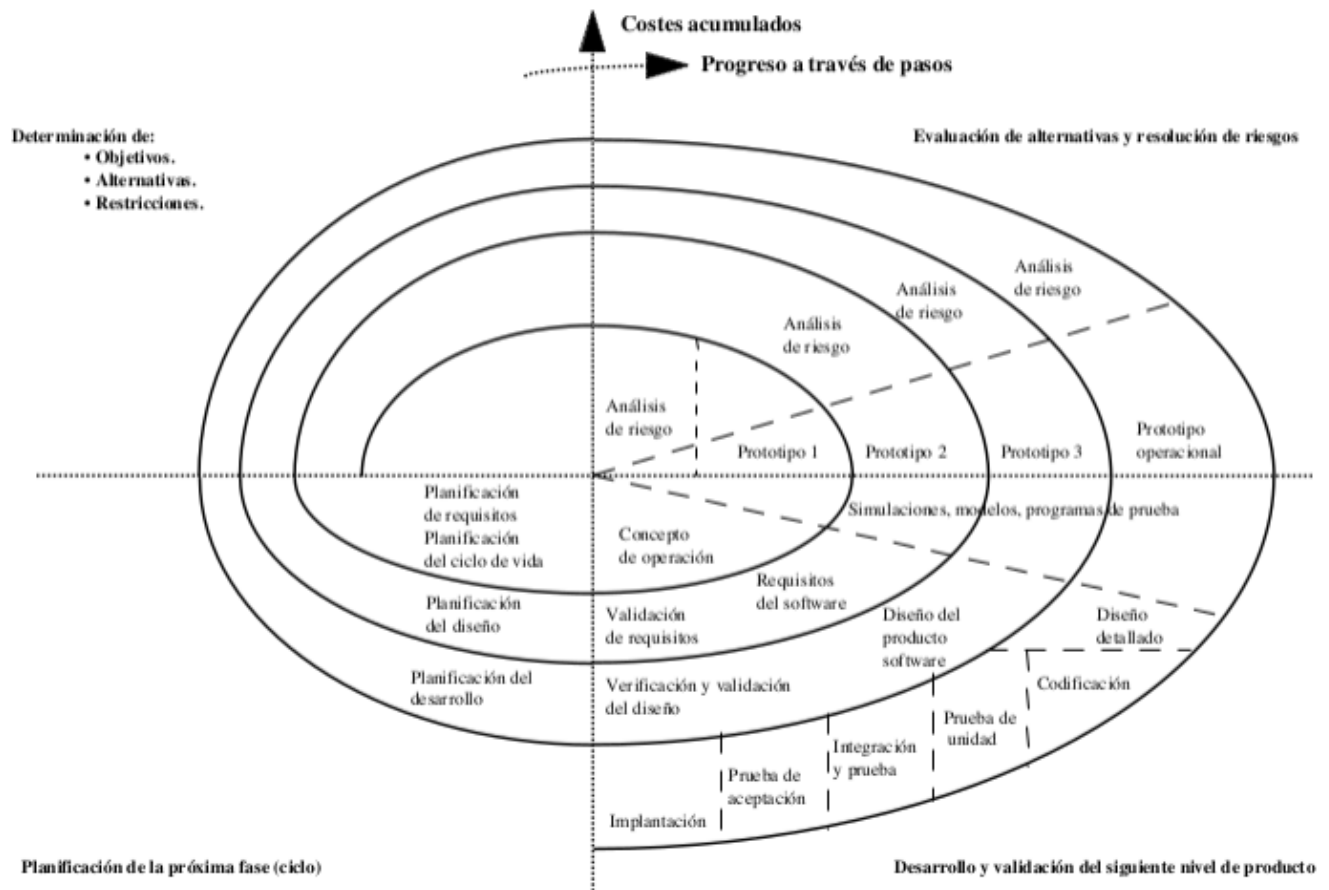
Inconvenientes:

- Puede ser difícil llegar al primer incremento.
- Es difícil establecer y priorizar las nuevas funcionalidades.
- Los incrementos anteriores pueden no ser válidos a posteriori.
- No sirve para proyectos cortos.

Modelo en espiral

Consiste en desarrollar versiones incrementales, pero partiendo de versiones prototipadas hasta versiones completas. Permite combinar la construcción de prototipos y la metodología cascada.

Enfatiza en el análisis de riesgos. La idea es que cada "vuelta de la espiral" determine unos objetivos, considere los riesgos y alternativas, desarrolle el siguiente nivel y deje planificada la siguiente fase. Sería de esta manera:



Ventajas:

- Permite adaptar el proceso de desarrollo a los cambios.
- Enlaza análisis de riesgos y prototipos.
- Permite reducir los riesgos antes de que se conviertan en problemas mayores.

Inconvenientes:

- Requiere de una buena consideración de riesgos.