

## Modelo en Cascada

Fortalezas	Oportunidades	Debilidades	Amenazas
<ul style="list-style-type: none"> <li>• Es el modelo clásico</li> <li>• Es intuitivo y simple de seguir</li> <li>• Los costes se pueden definir al principio del proyecto</li> </ul>	<p>La visualización es simple, lo que puede permitir dar un plazo de tiempo claro</p>	<ul style="list-style-type: none"> <li>• No refleja correctamente el proceso que se para siguiendo en el desarrollo de un proyecto, ya que requiere que todo esté completamente definido al principio.</li> <li>• El usuario final se mantiene al margen hasta que está finalizado el producto</li> </ul>	<p>En ocasiones, los fallos no se detectan hasta el final del proceso</p>

## Modelo en Incremental

Fortalezas	Oportunidades	Debilidades	Amenazas
<ul style="list-style-type: none"> <li>• Acomoda de forma fácil los cambios</li> <li>• Baja el tiempo de desarrollo inicial, ya que implementa funcionalidad parcial.</li> </ul>	<ul style="list-style-type: none"> <li>• Funciona como el modelo en cascada, pero permitiendo la retroalimentación.</li> <li>• Contiene menos riesgos que el modelo en cascada</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere de mucha planificación.</li> <li>• Requiere de metas claras</li> </ul>	<p>Puede llegar a ser problemático para sistemas en tiempo real y de procesamiento distribuido.</p>

## Modelo en Prototipos

Fortalezas	Oportunidades	Debilidades	Amenazas
Es perfecto para cuando el cliente tiene la idea general de lo que quiere en el sistema pero no tiene planeados aún los detalles.	Tiene una muy buena adaptabilidad en el sistema.	El usuario puede generarse sobre expectativas por prototipos con poca funcionalidad y mucha estética.	Es fácil tomar decisiones erróneas por estar haciendo implementaciones parciales.

## Modelo en Espiral

Fortalezas	Oportunidades	Debilidades	Amenazas
<ul style="list-style-type: none"><li>• Aprovecha la iteratividad del modelo de prototipos con la estructura que tiene el modelo en cascada.</li><li>• Reduce los riesgos del proyecto</li><li>• Integra el desarrollo con el mantenimiento</li></ul>	Se pueden tomar en cuenta mejoras y nuevos requisitos sin afectar el proceso.	Toma mucho tiempo en el desarrollo del sistema	Puede llegar a tener costos muy altos.

## Modelo basado en Componentes

Fortalezas	Oportunidades	Debilidades	Amenazas
El modularizar el proyecto permite separarlo de forma clara en varios equipos, y permite tener una visión clara de el proceso que se tiene que seguir	Se puede aprovechar mejor el tiempo al separar los equipos.	Para proyectos grandes se necesitan muchos recursos humanos para organizar los equipos y mantener la comunicación.	Las dependencias entre componentes puede causar problemas de organizacion.