

# Actividad 1 Unidad 2

Pablo Sanchez (21001135)

Modelo	Ventajas	Desventajas
Codificar y corregir	<ul style="list-style-type: none"><li>Es el mas rápido para problemas pequeños</li></ul>	<ul style="list-style-type: none"><li>Luego de varias correcciones, el código se hace difícil de leer</li><li>Software hecho de esta forma es frecuentemente rechazado por los clientes</li><li>Es difícil hacer pruebas en el código</li></ul>
Modelo en cascada	<ul style="list-style-type: none"><li>Es sencillo de entender</li><li>Es perfecto cuando se tienen de entrada todos los requisitos</li></ul>	<ul style="list-style-type: none"><li>Cada una de las iteraciones son costosas, aun mas si tienen que repetirse</li><li>Es muy probable que el software no cumpla con los requisitos, ya que pasa una gran cantidad de tiempo desde que se recolectan hasta que se entrega el software al usuario.</li><li>Es bastante inflexible a la hora de incorporar requisitos nuevos</li></ul>
Desarrollo evolutivo	<ul style="list-style-type: none"><li>Obtiene una rápida retroalimentacion del usuario</li><li>La especificación se hace de forma creciente</li><li>Cumple con las necesidades inmediatas del cliente</li></ul>	<ul style="list-style-type: none"><li>No es efectivo si el sistema se tiene que desarrollar rápido, ya que se tienen que producir documentos que reflejen cada versión del sistema</li><li>Los cambios continuos pueden ser perjudiciales para la estructura del software</li></ul>
Desarrollo formal de sistemas	<ul style="list-style-type: none"><li>Es atractivo sobre todo para sistemas donde hay requisitos de seguridad y confiabilidad importantes.</li><li>Es mejor para trabajar en proyectos grandes</li></ul>	<ul style="list-style-type: none"><li>Requiere de desarrolladores especializados y experimentados en el proceso para que se pueda llevar a cabo correctamente</li></ul>
Desarrollo basado en reutilización	<ul style="list-style-type: none"><li>Disminuye el costo y esfuerzo</li><li>Reduce el tiempo de entrega</li><li>Disminuye los riesgos</li></ul>	<ul style="list-style-type: none"><li>Puede que el software no termine de cumplir con las necesidades del cliente</li><li>Los componentes que se adquieren no estan en manos de los desarrolladores</li></ul>
Desarrollo Incremental	<ul style="list-style-type: none"><li>Los clientes no esperan hasta el final de desarrollo para empezar a usar el software</li><li>Los clientes pueden aclarar requisitos que no tengan claros conforme ven el sistema</li><li>Las partes mas importantes se entregan primero, lo que permite que se hagan mas pruebas con estos módulos, disminuyendo el riesgo de fallos</li></ul>	<p>Se tiene que asegurar de que cada incremento:</p> <ul style="list-style-type: none"><li>Es pequeño para limitar los riesgos</li><li>Incremente la funcionalidad</li></ul> <p>Ademas de que puede ser complicado establecer las correspondencias entre requisitos e incrementos</p>
Desarrollo en espiral	Este considera explicitamente los riesgos, reduciéndolos considerablemente.	Puede ser muy complicado y ocupar mucho tiempo en proyectos pequeños.