
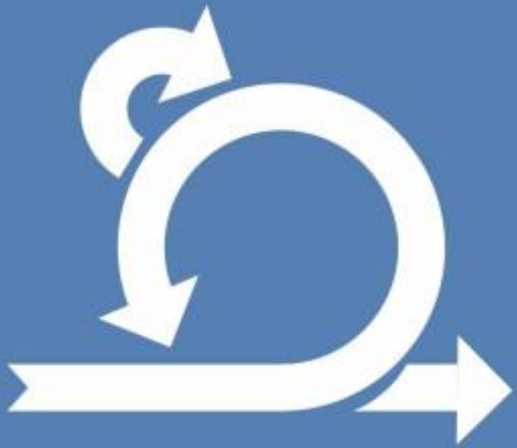


Arquitectura del Software

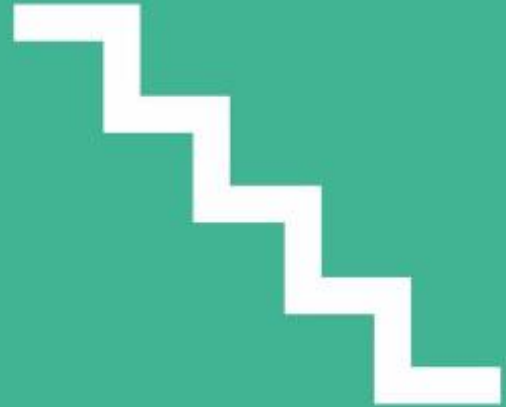


AGILE

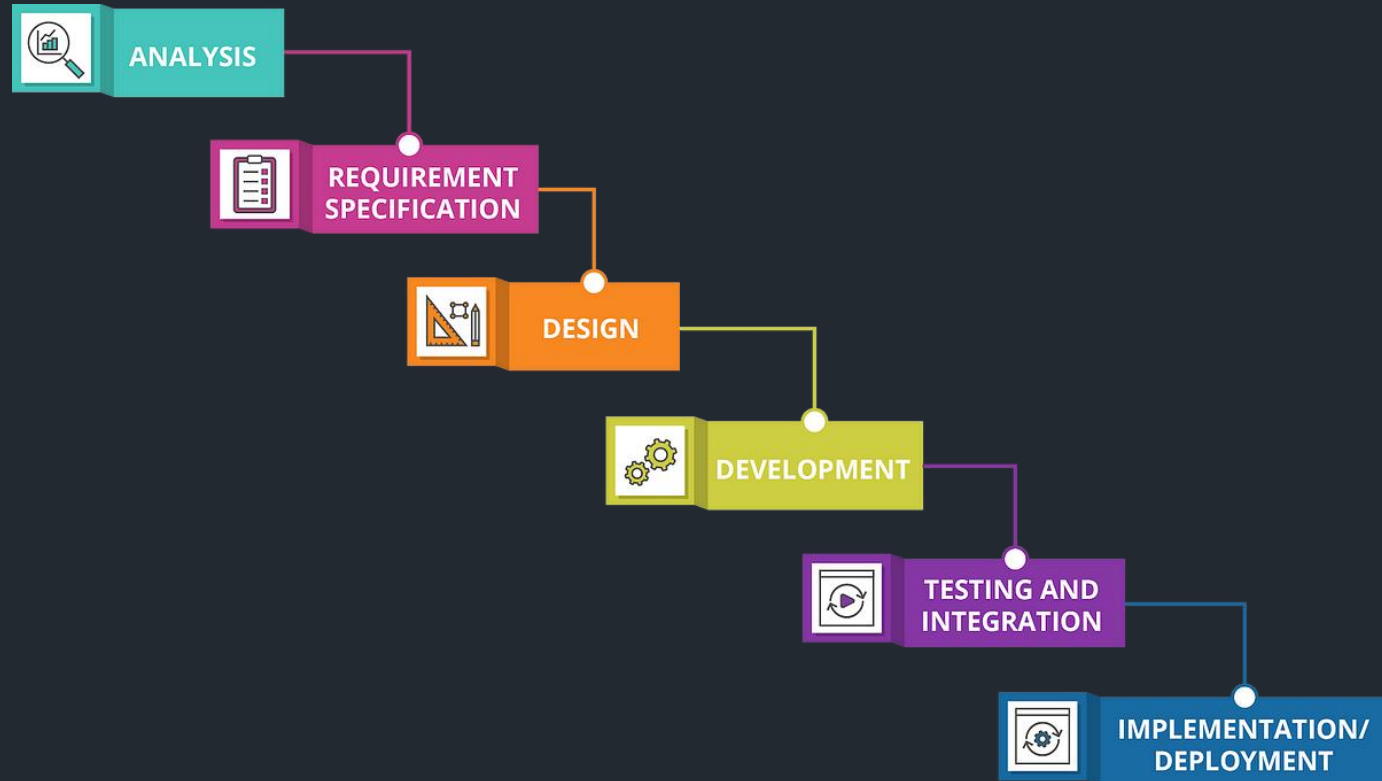


VS

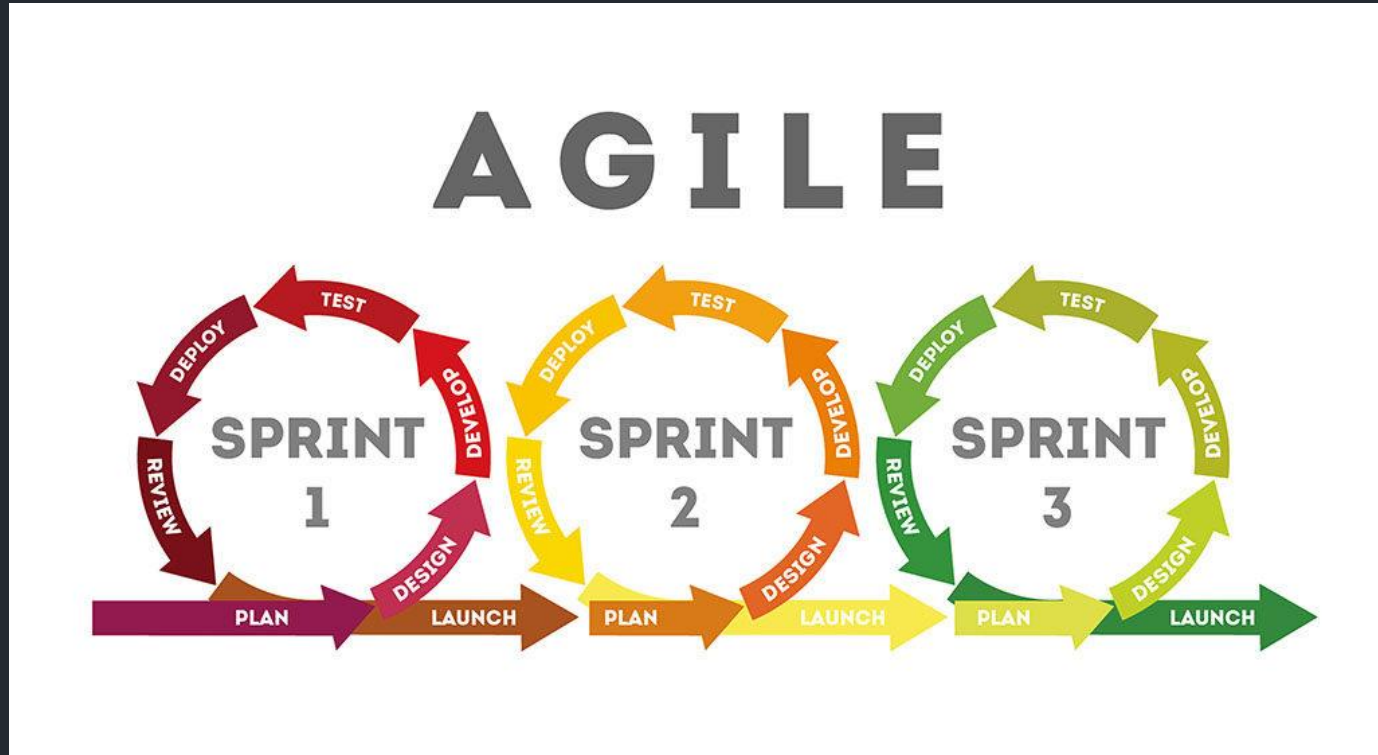
WATERFALL



● Arquitectura del Software – Modelos SDLC - **Cascada**

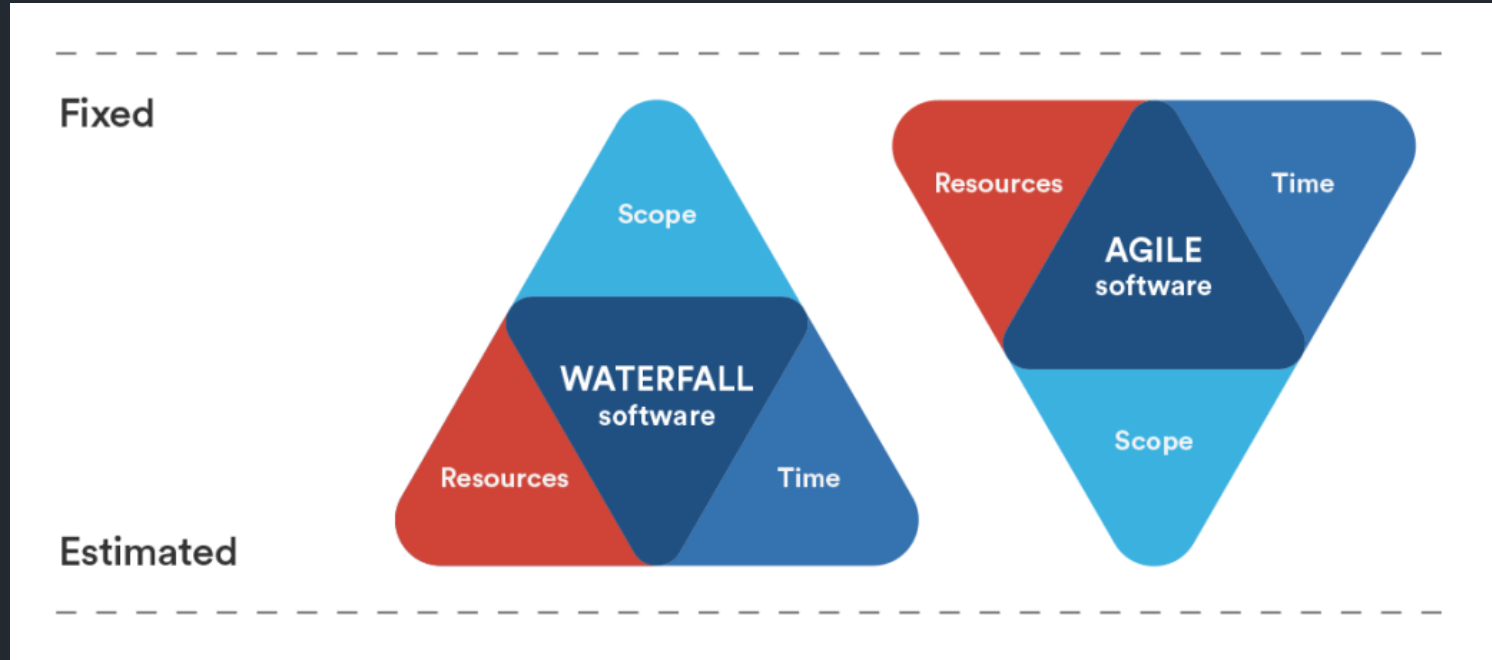


- Arquitectura del Software – Modelos SDLC - Ágil



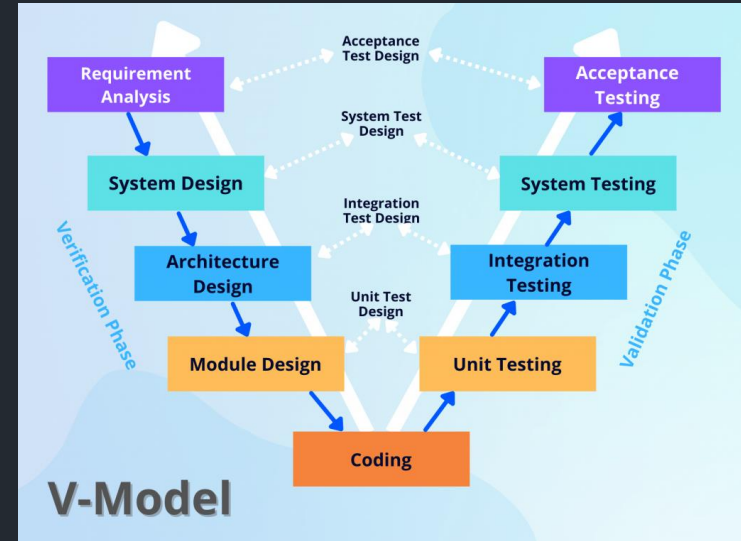
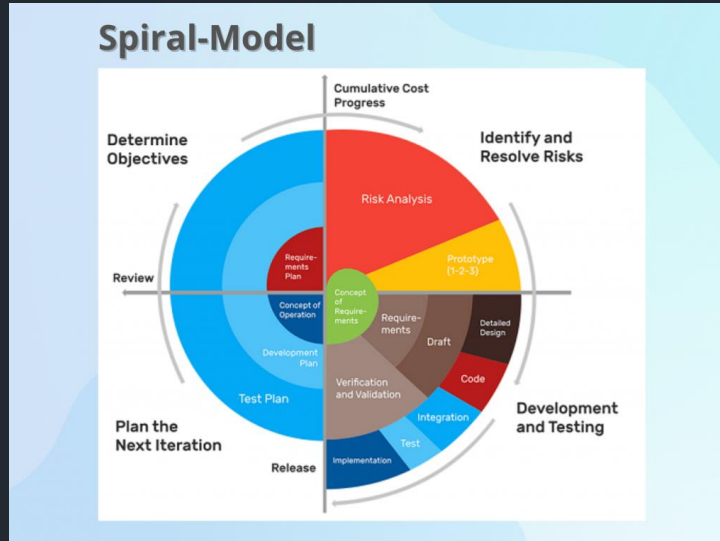
Característica	Cascada	Ágil
Gestión de alcance y recursos	Alcance fijo con recursos y tiempo estimados.	Recursos y tiempo fijos con alcance estimado.
Ciclo de desarrollo de productos	Producto entregado al final con un enfoque lineal y por fases.	Producto entregado de forma incremental con ciclos de desarrollo iterativos a corto plazo.
Planificación de la gestión de proyectos	Planificación detallada del proyecto a largo plazo completada antes de la ejecución.	Planificación continua basada en iteraciones.
Composición del equipo	Roles de equipo definitivos con asignaciones individuales de responsabilidad.	Roles de equipo flexibles y multifuncionales en los que el equipo comparte la responsabilidad por igual
Participación de las partes interesadas	Las partes interesadas suelen participar al principio y al final del desarrollo del proyecto.	Las partes interesadas están profundamente involucradas durante todo el desarrollo del proyecto.

- Arquitectura del Software – Modelos SDLC



Arquitectura del Software – Modelos SDLC

Otros modelos SDLC

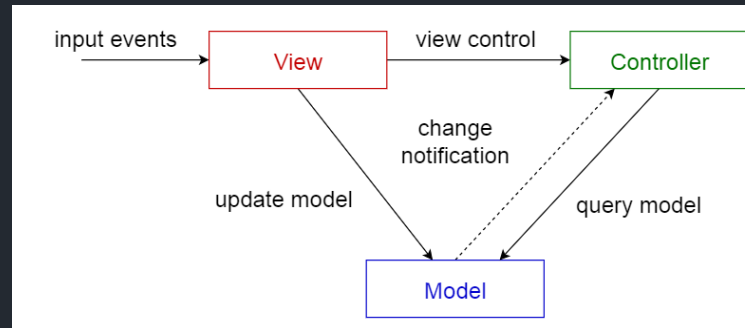


● Arquitectura del Software – Patrones Arquitectónicos

- Un patrón arquitectónico es una solución general y reutilizable a un problema común en la arquitectura de software dentro de un contexto dado.
- Los patrones arquitectónicos son similares al patrón de diseño de software, pero tienen un alcance más amplio.
- Tipos
 - Modelo-Vista-Controlador (MVC)
 - Capas (Layered Architecture)
 - Cliente-Servidor
 - Otros

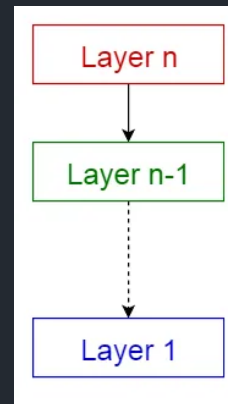
Arquitectura del Software – Patrones - MVC

- Separa una aplicación en tres componentes principales:
 - **Modelo:**
 - Gestiona los datos y la lógica de la aplicación
 - **Vista:**
 - Se encarga de la presentación y la interfaz de usuario
 - **Controlador:**
 - Maneja la interacción entre el Modelo y la Vista
- Promueve la modularidad y la escalabilidad de las aplicaciones.



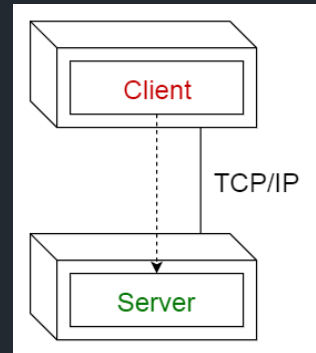
Arquitectura del Software – Patrones - Capas

- Divide la aplicación en capas o niveles lógicos, donde cada capa tiene una responsabilidad específica
- Las capas más comunes son:
 - Capa de Presentación o UI
 - Capa de Aplicación o de Servicio
 - Capa de Lógica de Negocio
 - Capa de Acceso a Datos



● Arquitectura del Software – Patrones – **Cliente-Servidor**

- Los sistemas se dividen en dos partes
 - Cliente:
 - Solicita servicios o recursos
 - Servidor
 - Proporciona esos servicios o recursos
- Puede haber múltiples clientes que se conectan a un servidor central.
- Este patrón es fundamental en aplicaciones web y sistemas distribuidos.



Enlaces de interés

- Documentación Agile: <https://www.atlassian.com/es/agile>
- Documentación MVC: <https://www.netmentor.es/entrada/explicacion-mvc>

