



**Universidad
Andrés Bello®**

Conectar • Innovar • Liderar



Universidad
Andrés Bello®

Facultad de Ingeniería



Ingeniería de Software I

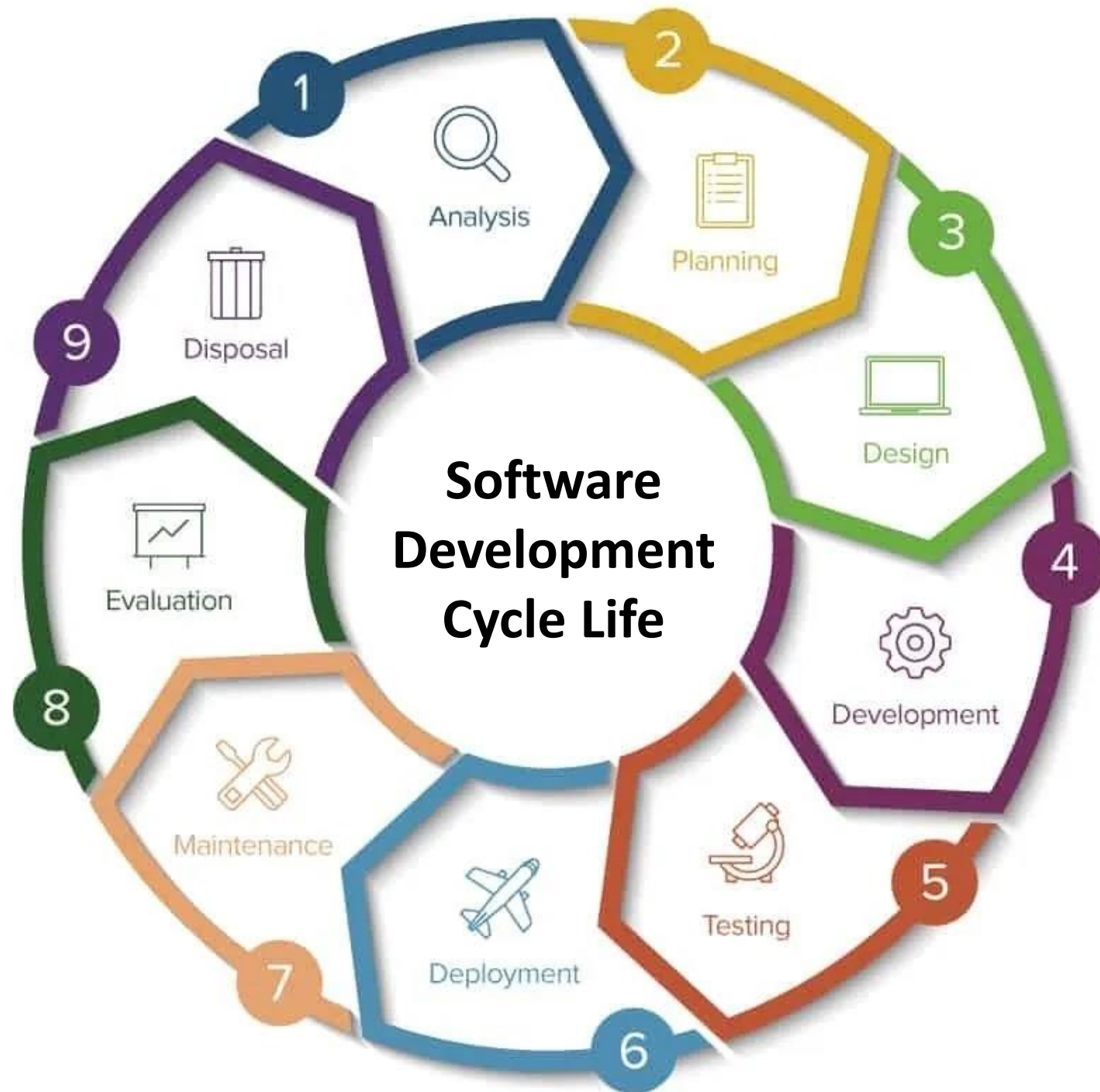
Ismael Moreno Flores



Ingeniería de Software I

Unidad I: Fundamentos de la Ingeniería de Software

Ismael Moreno Flores





Objetivos:



Al término de la experiencia de hoy debemos estar en la capacidad de:

- Diferenciar los enfoques de las metodologías tradicionales de los marcos de trabajo ágiles para el desarrollo de software.



Contenidos:

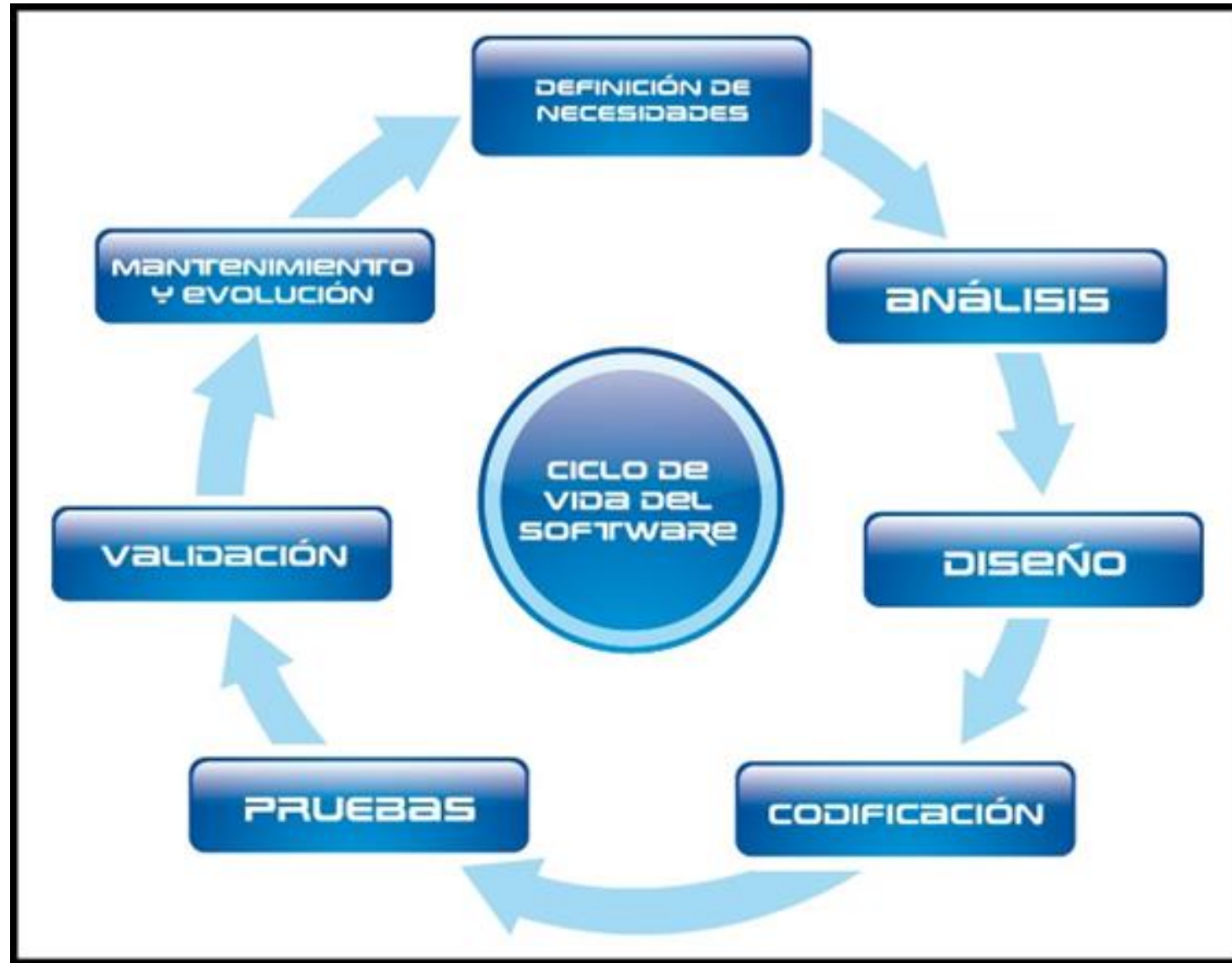


- ¿Cuáles son los modelos de desarrollo de software?
- ¿Cuáles son las características de las metodologías tradicionales?
- ¿Cuáles son los marcos de trabajo ágiles?



La idea de aplicar
**ingeniería de
software** es
garantizar la **calidad
del producto.**





Las 7 fases del SDCL





Primero se usaron
modelos de
desarrollo y luego
se usaron
metodologías de
desarrollo



Modelos o enfoques de desarrollo de software

Modelos de proceso prescriptivos:

- Cascada
- Incremental
- Prototipos
- Espiral

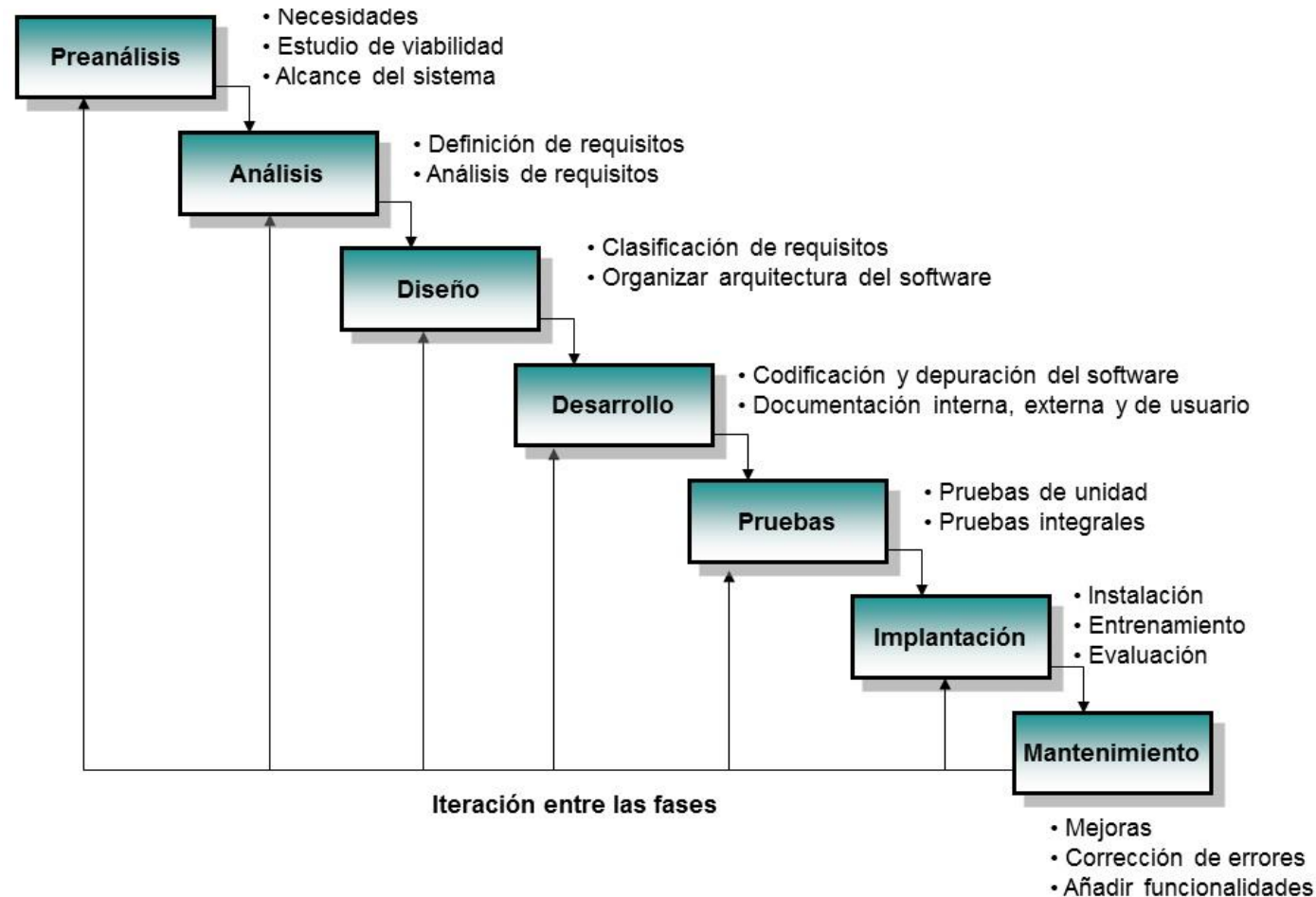
Modelos de proceso especializados:

- Desarrollo basado en componentes
- Métodos formales
- Software orientado a aspectos



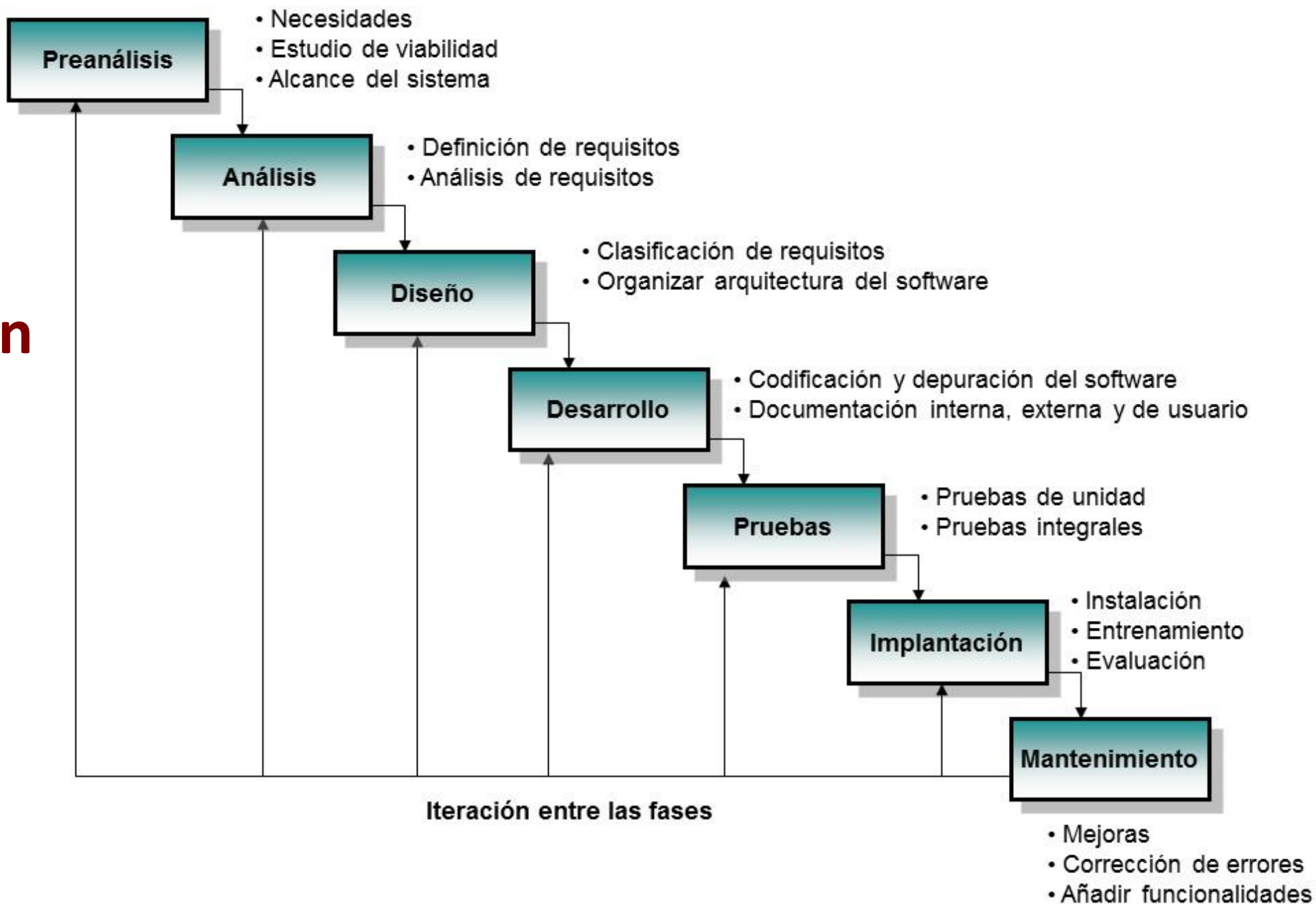
Modelos o enfoques de desarrollo de software

Modelo en Cascada



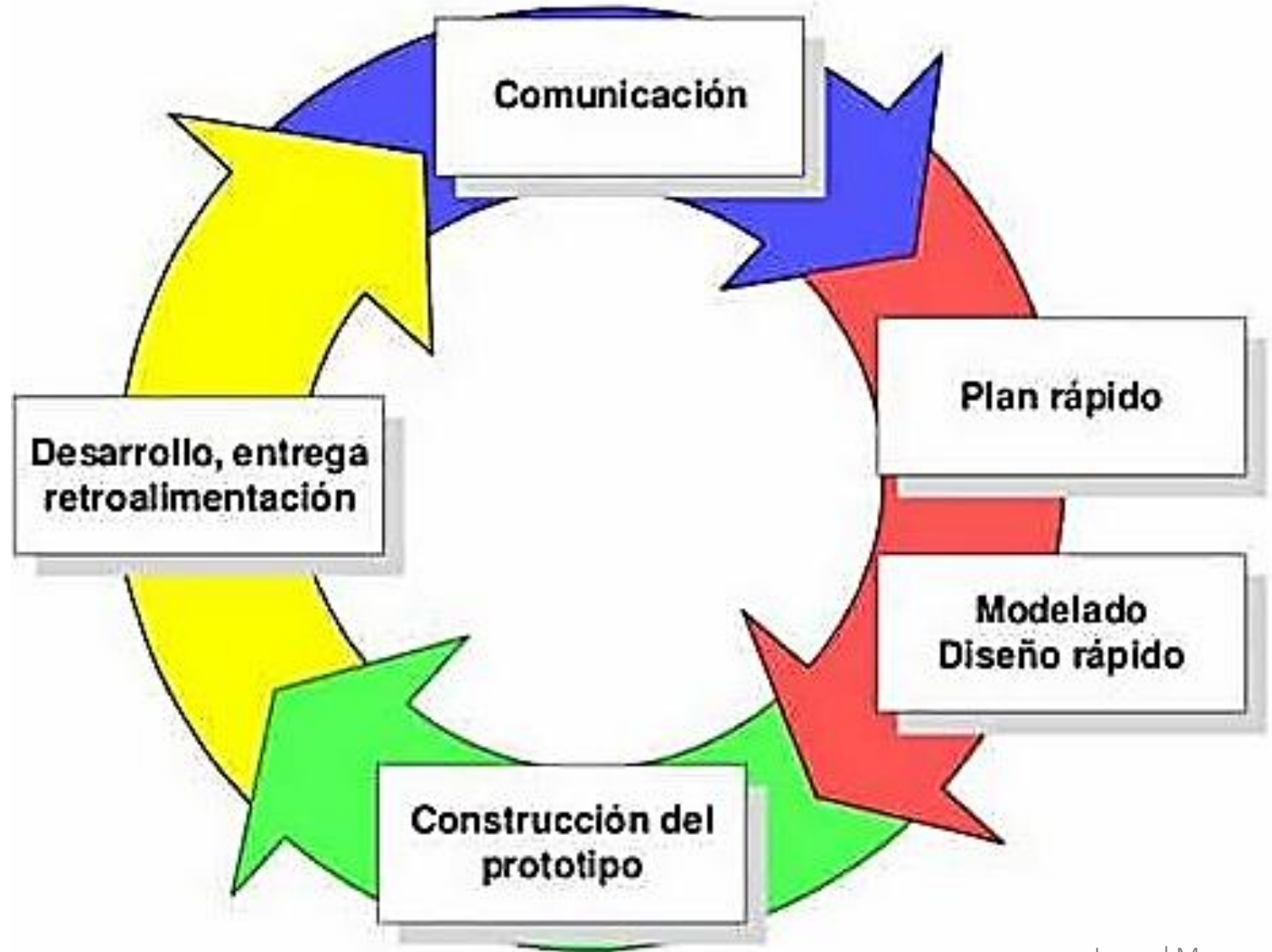


Modelo en Cascada



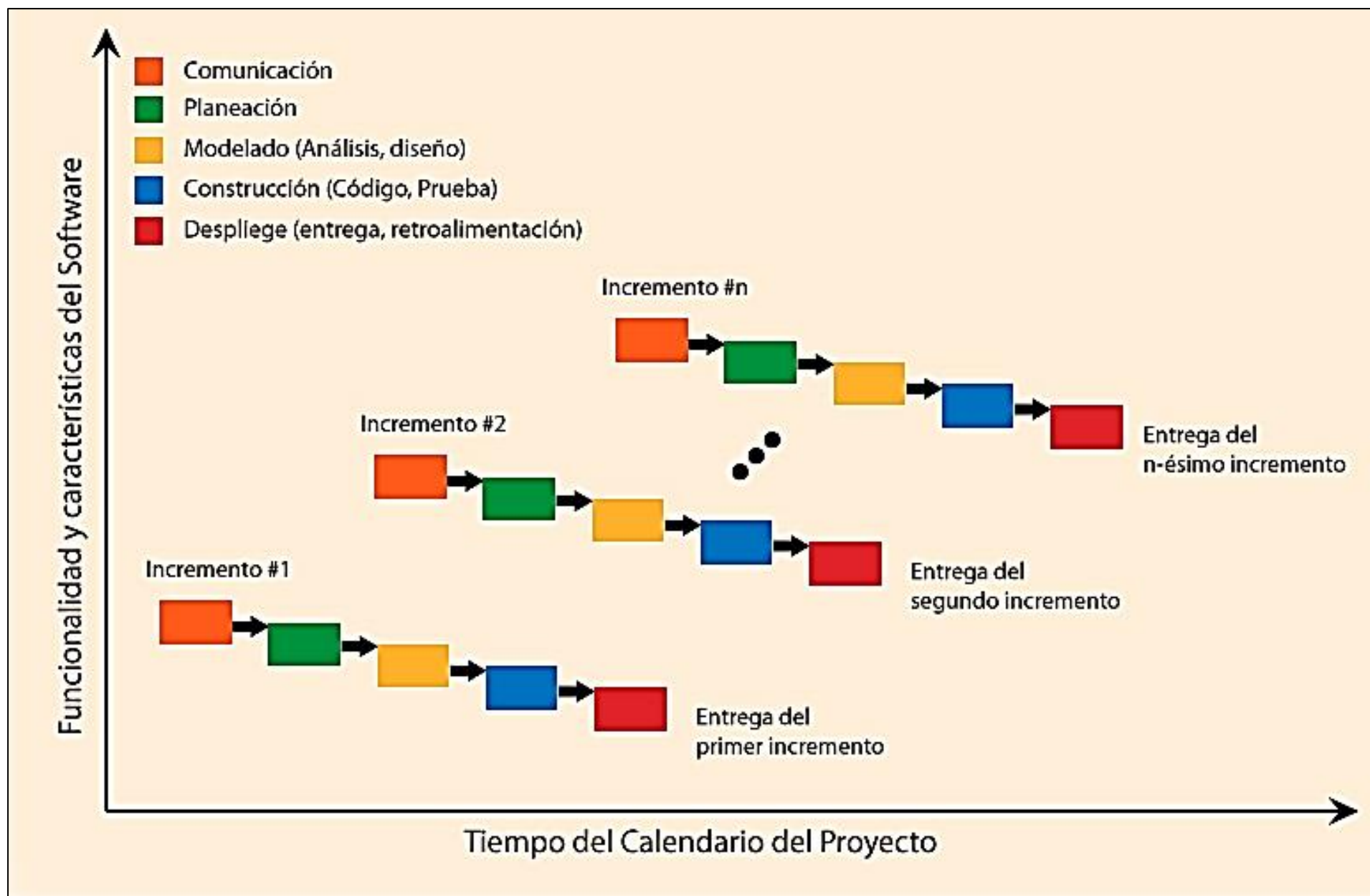


Modelo por prototipos



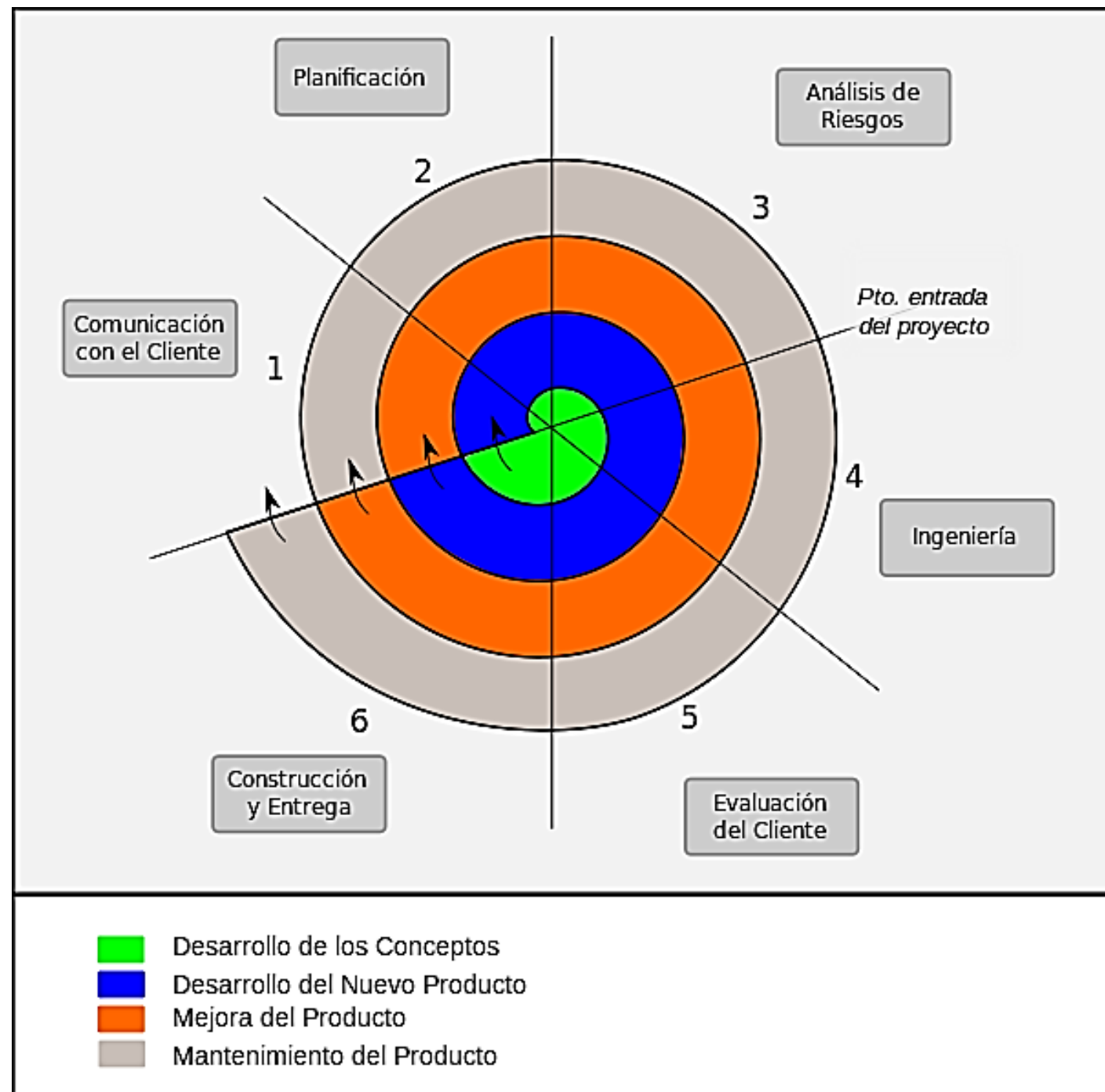


Modelo Incremental



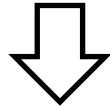


Modelo en espiral

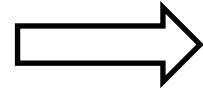


Manifiesto Ágil

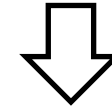
Priorizar este aspecto



Personas e interacciones

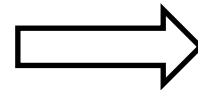


Sobre este aspecto



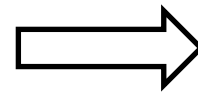
Procesos y herramientas

Software funcional



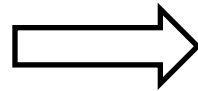
Documentación exhaustiva

Colaboración con el cliente



Negociación contractual

Respuesta ante el cambio



Seguimiento de un plan



12 principios del Manifiesto Ágil

<p>Satisfacer al usuario</p>  <p>Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.</p>	<p>Aceptamos los cambios</p>  <p>Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.</p>	<p>Entregas frecuentes</p>  <p>Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.</p>	<p>Negocio e IT juntos</p>  <p>Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.</p>
<p>Personas motivadas</p>  <p>Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.</p>	<p>Comunicación cara a cara</p>  <p>El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.</p>	<p>Producto funcionando</p>  <p>El software funcionando es la medida principal de progreso.</p>	<p>Ritmo sostenible</p>  <p>Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.</p>
<p>Excelencia técnica</p>  <p>La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.</p>	<p>Sencillez</p>  <p>La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.</p>	<p>Arquitectura y diseño</p>  <p>Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.</p>	<p>El equipo reflexiona</p>  <p>A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.</p>

Fuente: <https://agilemanifesto.org/iso/es/principles.html>

Infografía realizada por @VallejoAngeles y @alfredovela de ticsyformacion.com



Metodologías tradicionales

(Pesadas,
Disciplinadas,
Estructuradas,
Formales)

Marcos de trabajo ágiles

(Livianos,
Flexibles,
Adaptativos,
Informales)



Metodologías tradicionales

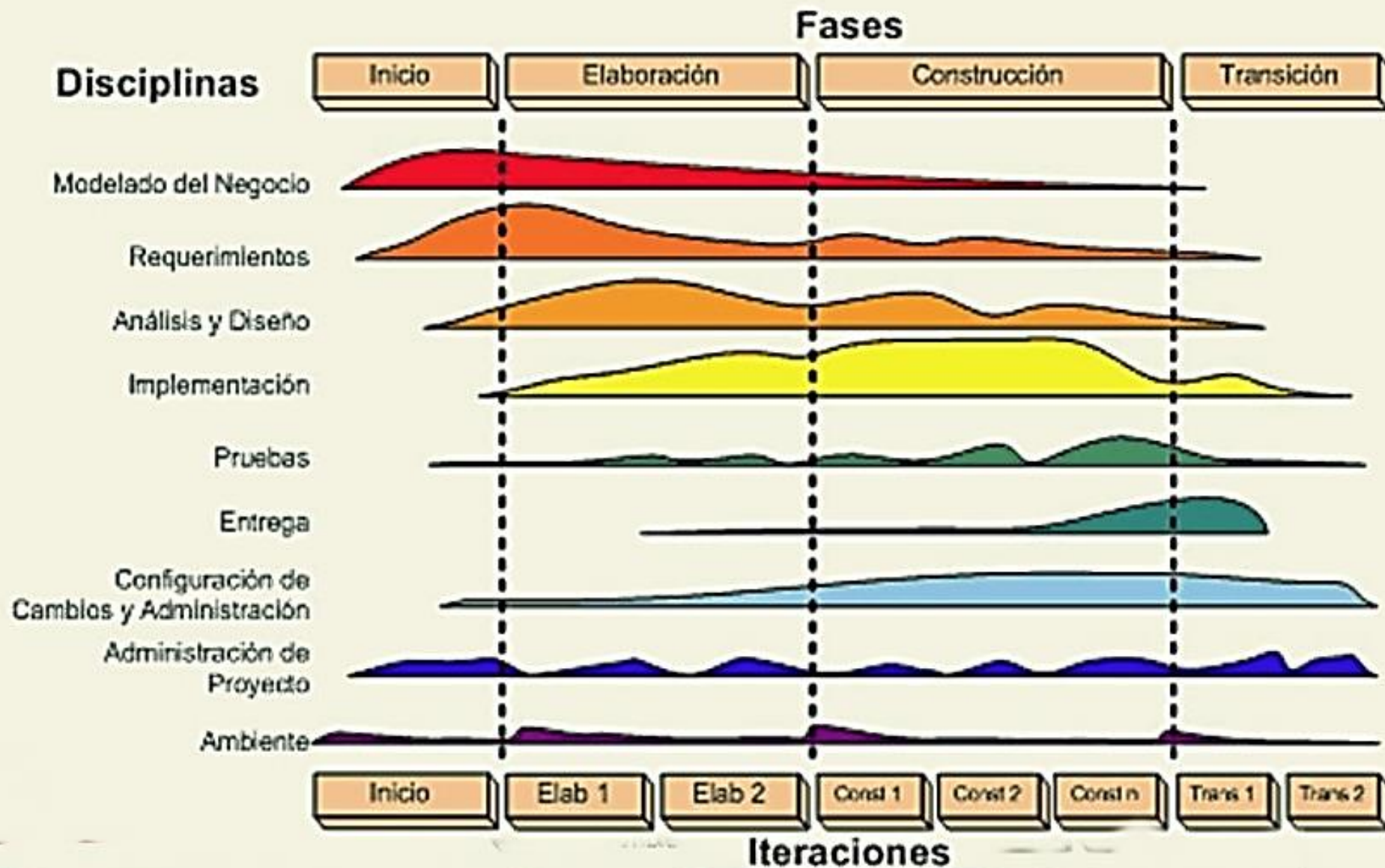
- Proceso Racional Unificado (RUP: Rational Unified Process)
 - Entorno de Soluciones de Microsoft (MFS: Microsoft Framework Solutions)
- Metodología Waterfall.

Marcos de trabajo ágiles

- Scrum
- Programación Extrema (XP: eXtreme Programming)
- Kanban
- Crystal
- LSD (Lean Software Development)



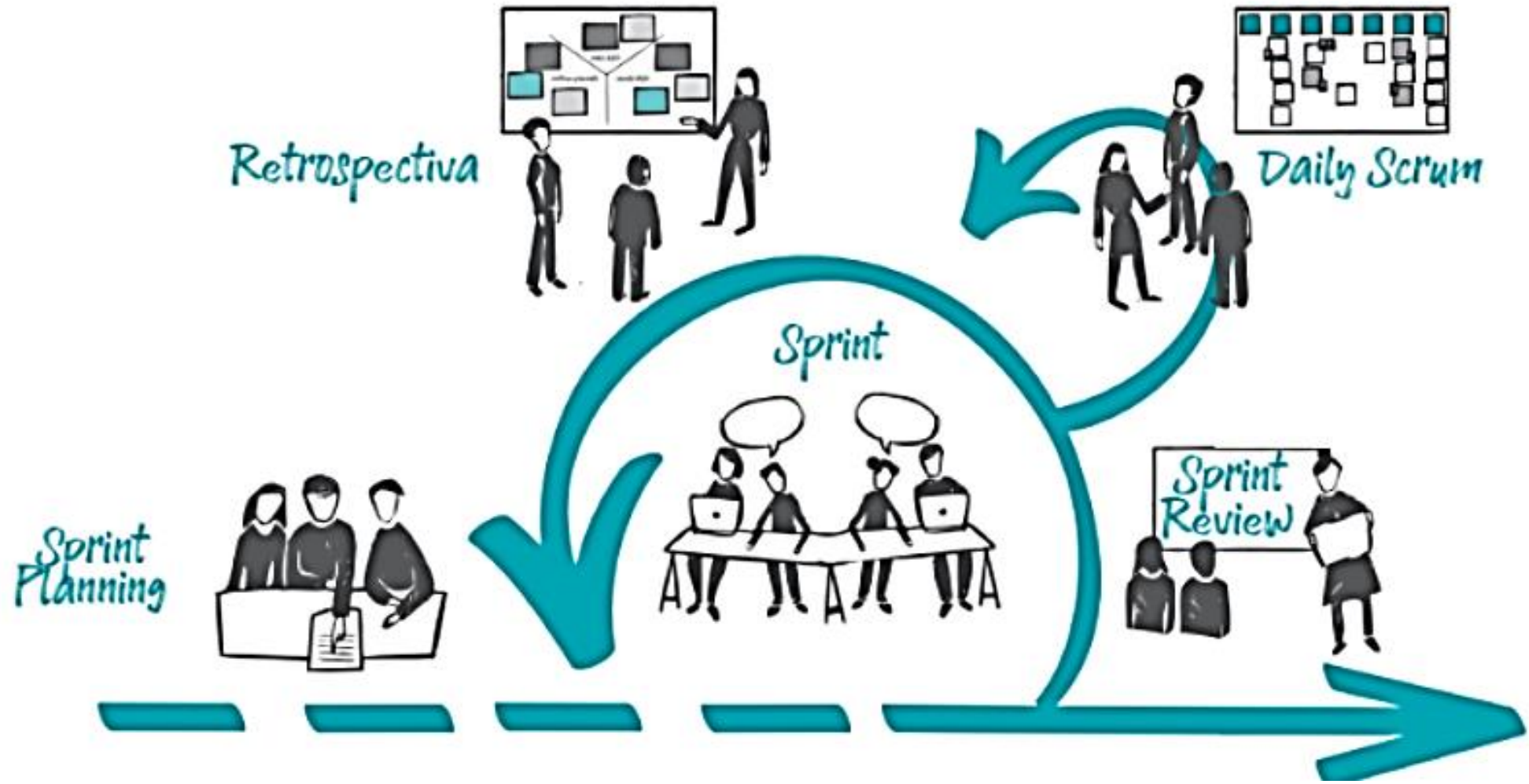
RUP





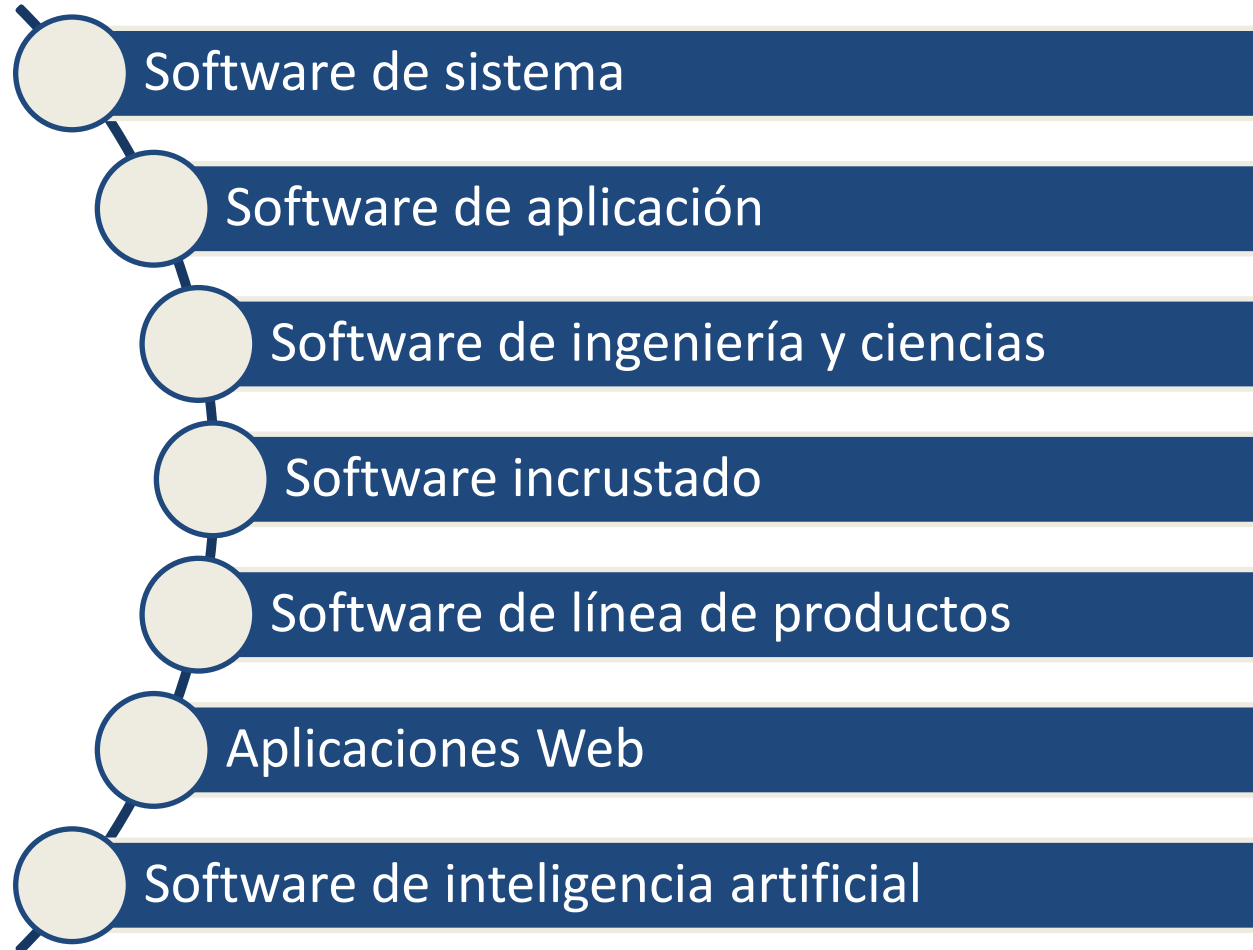
Metodología SCRUM







Dominios de aplicación del Software



(Pressman, 2015)

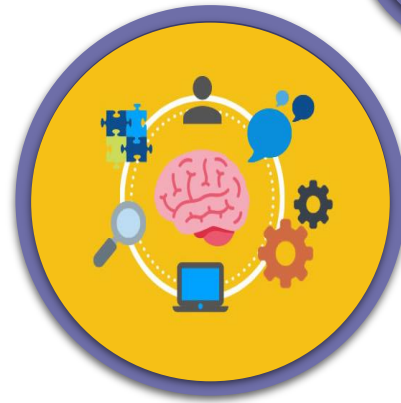
Ingeniería de Software

Es el establecimiento y uso de principios fundamentales de la ingeniería con objeto de desarrollar en forma económica software que sea confiable y que trabaje con eficiencia en máquinas reales.

(Bauer en Pressman, 2015)



Ingeniería de Software I



Implementar un proceso para el desarrollo de un producto tecnológico, guiado por **metodologías formales** de la Ingeniería de Software y aplicando herramientas y técnicas que garanticen la eficiencia del proceso.

Las 7 fases del SDCL





Metodologías tradicionales

- Proceso Racional Unificado (RUP: Rational Unified Process)
 - Entorno de Soluciones de Microsoft (MFS: Microsoft Framework Solutions)
- Metodología Waterfall.

Marcos de trabajo ágiles

- Scrum
- Programación Extrema (XP: eXtreme Programming)
- Kanban
- Crystal
- LSD (Lean Software Development)

🔍 ¿En qué puedo profundizar?

Actividades
recomendadas
para el trabajo
autónomo



🚧 ¿En qué puedo profundizar?

Cuáles son las características generales de los siguientes

Modelos de proceso especializados:

- Desarrollo basado en componentes
- Métodos formales
- Software orientado a aspectos



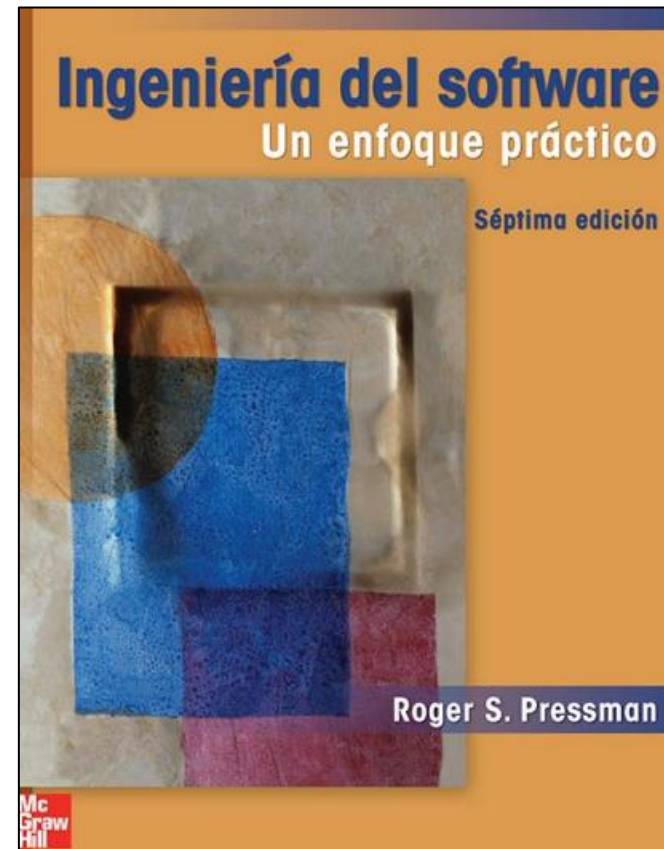
🏠 ¿En qué puedo profundizar?

Lectura sugerida:

EL PROCESO DEL SOFTWARE 25

CAPÍTULO 2 Modelos del proceso 26

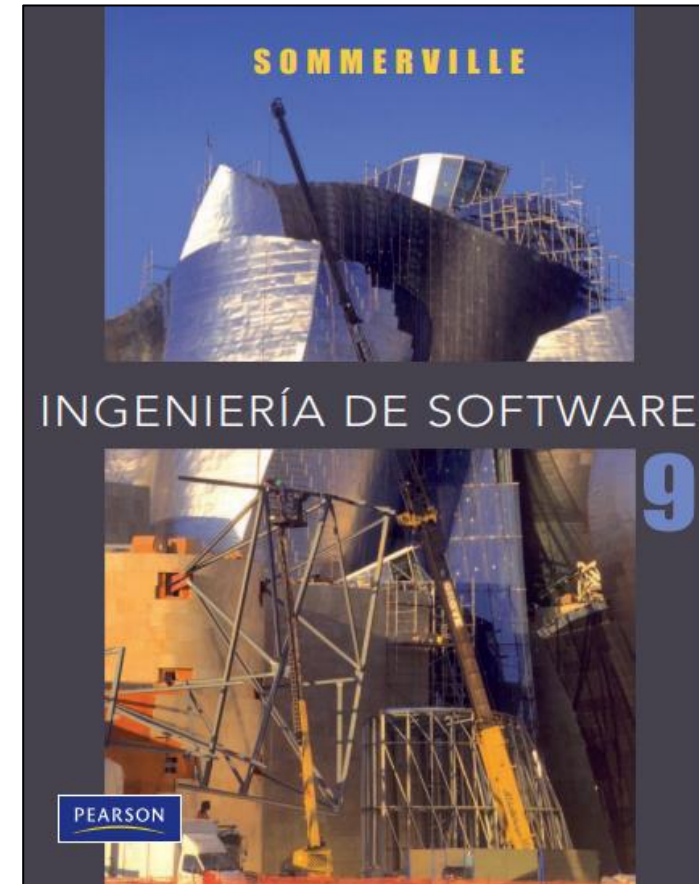
CAPÍTULO 3 Desarrollo ágil 55



🔍 ¿En qué puedo profundizar?

Lectura sugerida:

Parte 1	Introducción a la ingeniería de software	1
Capítulo 1	Introducción	3
Capítulo 2	Procesos de software	27
Capítulo 3	Desarrollo ágil de software	56





**Muchas
gracias por su
atención**





Recursos utilizados

Pressman, R. (2010). Ingeniería del Software: un enfoque práctico. 7ma Ed. México D.F.: Mc Graw-Hill Interamericana.

Sommerville, I. (2011). Ingeniería de Software. 9na Ed. México D.F.: Pearson Educación.