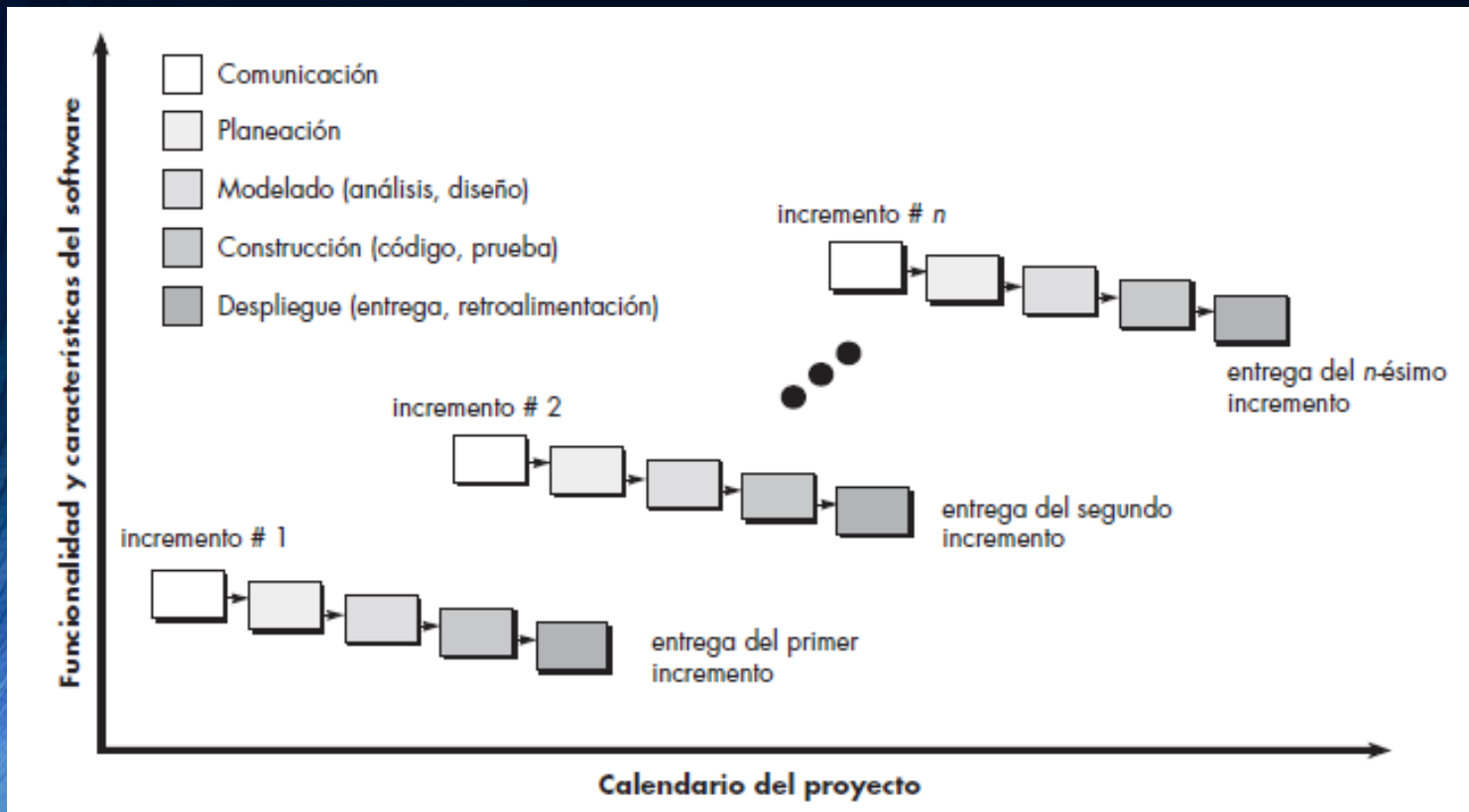


Modelos Iterativos Incrementales

POR MARIO ROSSAINZ LÓPEZ
INGENIERIA DE SOFTWARE II
OTOÑO 2022
NRC: 10838

Modelo de Proceso Incremental



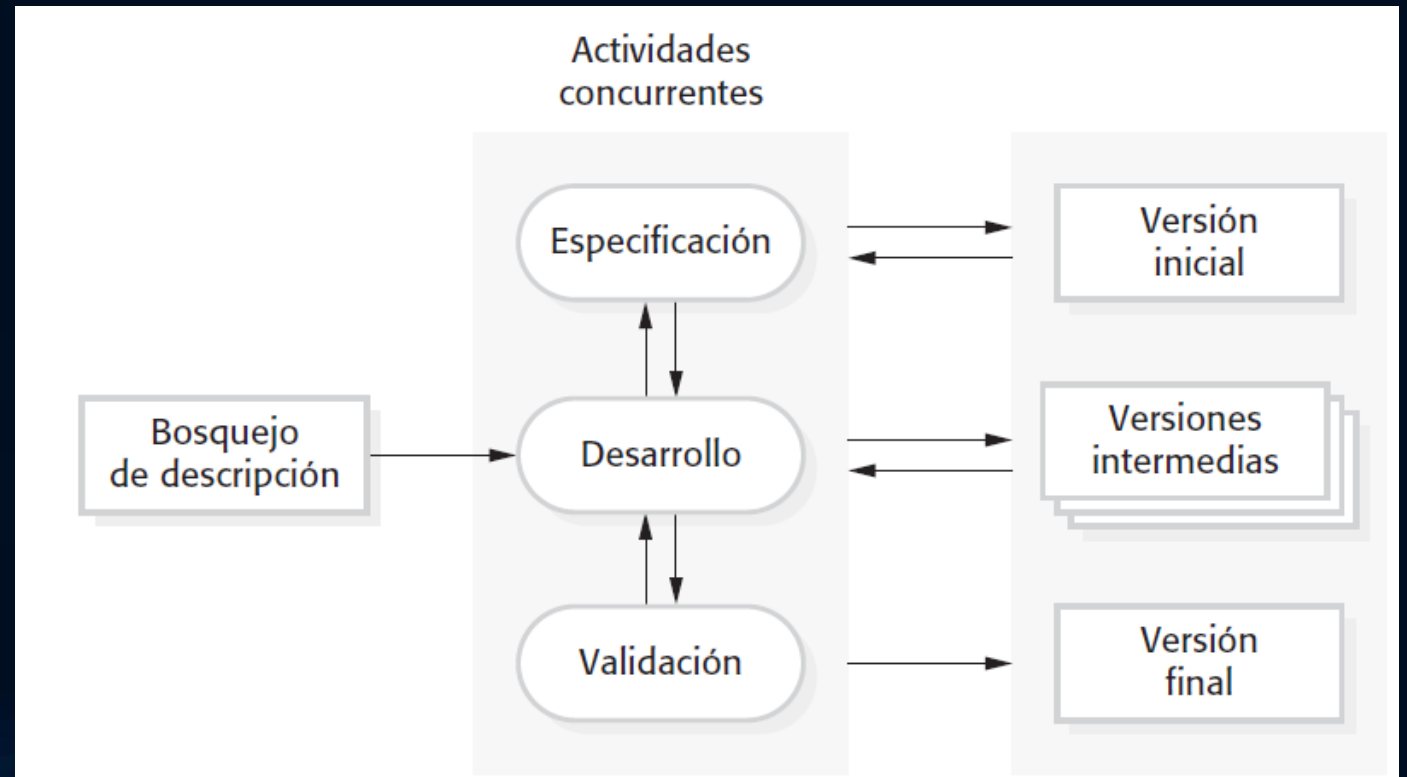
• *Combina elementos de los flujos de proceso lineal en paralelo aplicando secuencias lineales en forma escalonada a medida que avanza el calendario de actividades. Cada secuencia lineal produce "incrementos" de software susceptibles de entregarse.*

• El modelo de proceso incremental se centra en que en cada incremento se entrega un producto que ya opera.

Modelo de Proceso Incremental

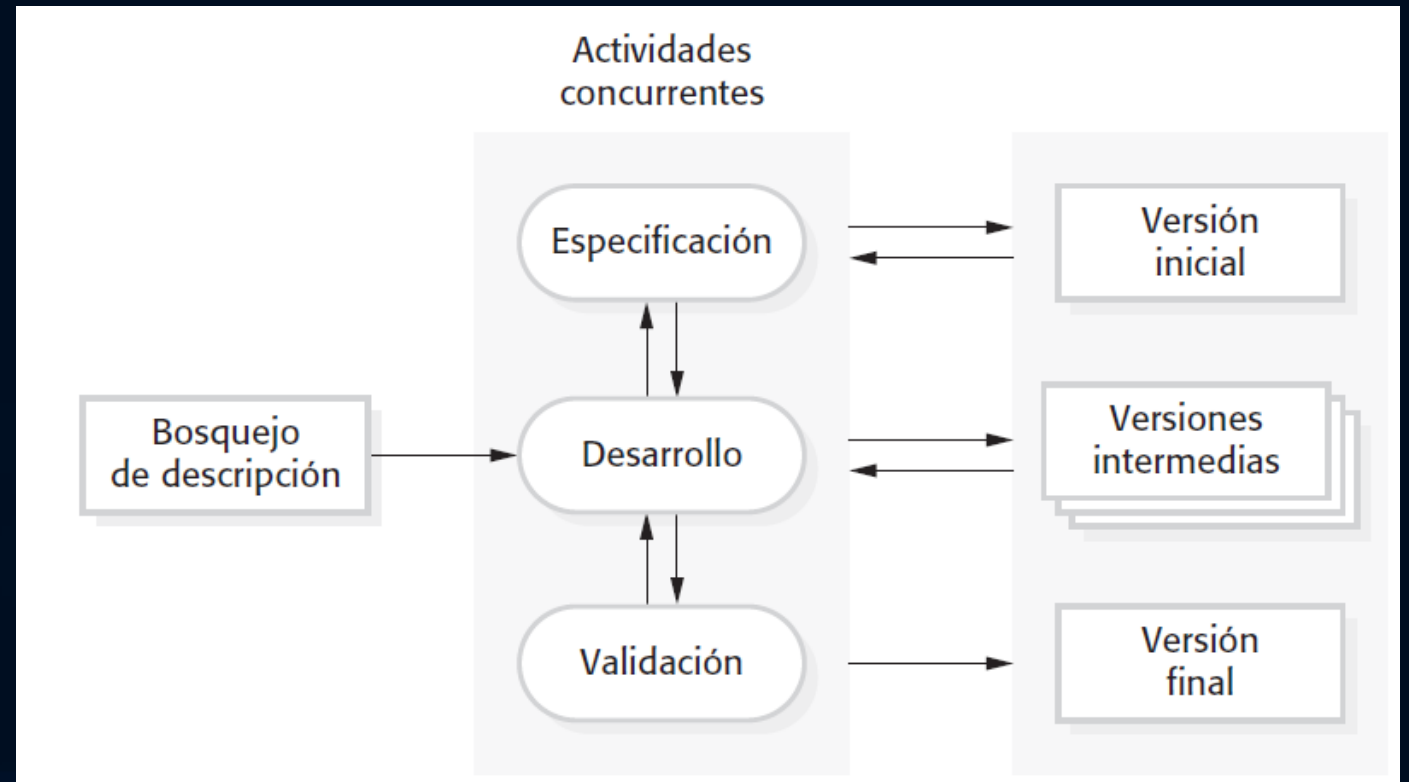
El desarrollo incremental se basa en la idea de:

- Diseñar una implementación inicial
- Exponer ésta al comentario del usuario, y luego
- Desarrollarla en sus diversas versiones hasta producir un sistema adecuado



Modelo de Proceso Incremental

- Las actividades de especificación, desarrollo y validación están entrelazadas en vez de separadas, con rápida retroalimentación a través de las actividades.
- El desarrollo de software incremental es mejor que un enfoque en cascada para la mayoría de los sistemas empresariales de comercio electrónico y personales



Modelo de Proceso Incremental

- El desarrollo incremental refleja la forma en que se resuelven problemas.
- Rara vez se trabaja por adelantado una solución completa del problema, más bien se avanza en una serie de pasos hacia una solución y se retrocede cuando se detecta que se cometieron errores.
- Al desarrollar el software de manera incremental, resulta más barato y fácil realizar cambios en el software conforme éste se diseña.



Modelo de Proceso Incremental

- Cada incremento o versión del sistema incorpora algunas de las funciones que necesita el cliente.
- Por lo general, los primeros incrementos del sistema incluyen la función más importante o la más urgente. Esto significa que el cliente puede evaluar el desarrollo del sistema en una etapa relativamente temprana, para constatar si se entrega lo que se requiere.
- En caso contrario, sólo el incremento actual debe cambiarse y, posiblemente definir una nueva función para incrementos posteriores

Modelo de Proceso Incremental

- VENTAJAS:

Se reduce el costo de adaptar los requerimientos cambiantes del cliente. La cantidad de análisis y la documentación que tiene que reelaborarse son mucho menores de lo requerido con el modelo en cascada.

Es más sencillo obtener retroalimentación del cliente sobre el trabajo de desarrollo que se realizó. Los clientes pueden comentar las demostraciones del software y darse cuenta de cuánto se ha implementado. Los clientes encuentran difícil juzgar el avance a partir de documentos de diseño de software.

Es posible que sea más rápida la entrega e implementación de software útil al cliente, aun si no se ha incluido toda la funcionalidad. Los clientes tienen posibilidad de usar y ganar valor del software más temprano de lo que sería posible con un proceso en cascada.

Modelo de Proceso Incremental

DESVENTAJAS:

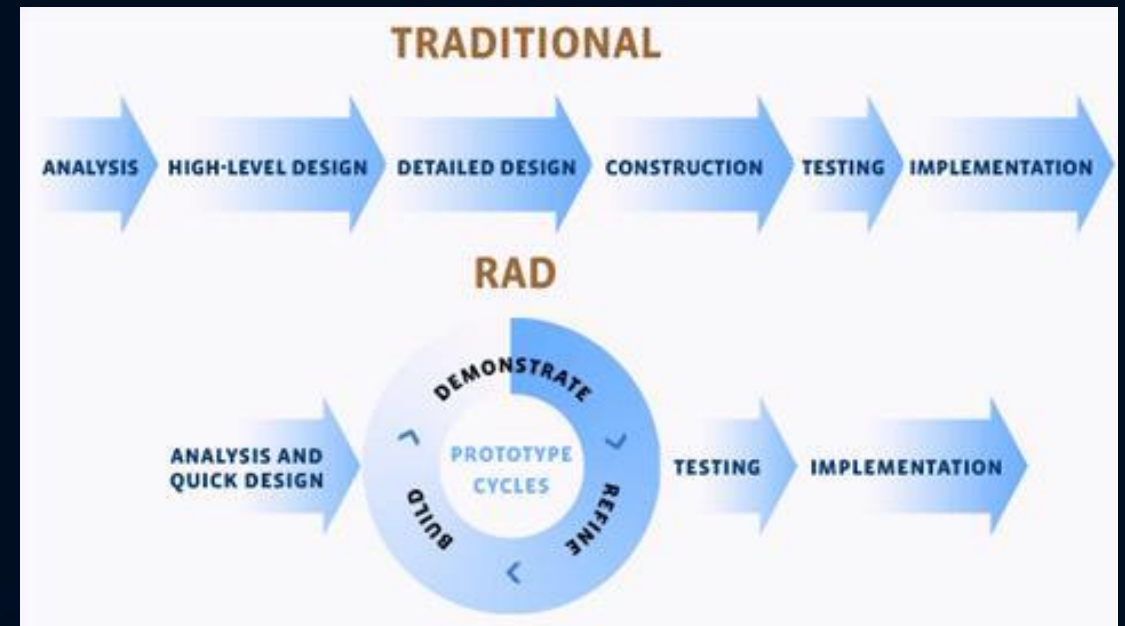
- El proceso no es visible. Los administradores necesitan entregas regulares para medir el avance. Si los sistemas se desarrollan rápidamente, resulta poco efectivo en términos de costos producir documentos que reflejen cada versión del sistema.
- La estructura del sistema tiende a degradarse conforme se tienen nuevos incrementos. A menos que se gaste tiempo y dinero en la refactorización para mejorar el software, el cambio regular tiende a corromper su estructura. La incorporación de más cambios de software se vuelve cada vez más difícil y costosa.

Modelo RAD (Rapid Application Development)

- El modelo RAD o modelo de Desarrollo Rápido de Aplicaciones es un enfoque al desarrollo de software dirigido a la entrega rápida del software.
- Con frecuencia implica el uso de programación de bases de datos y herramientas de soporte de desarrollo, como generadores de pantalla y reportes.
- Se vale muchas veces del uso de las herramientas CASE
- Fue creado por James Martin en 1991

Modelo RAD (Rapid Application Development)

- El modelo de desarrollo rápido de aplicaciones (RAD) pretende ser una técnica ágil de desarrollo de software que da prioridad a las entregas e iteraciones rápidas de prototipos.
- A diferencia de la metodología de cascada, RAD tiene más en cuenta el uso del software y la opinión del usuario que la planificación rigurosa y el registro de los requisitos.

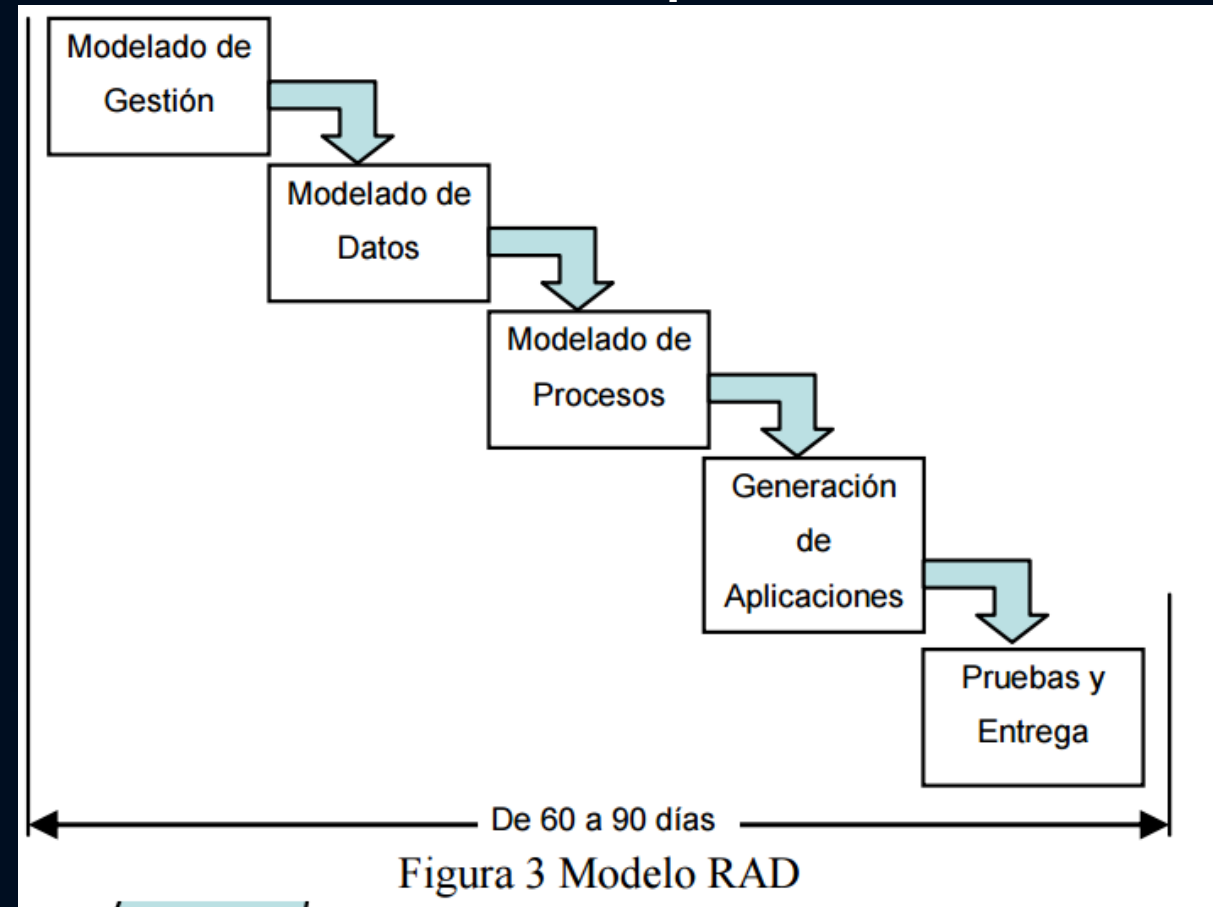


Modelo RAD (Rapid Application Development)

- VENTAJAS:
- Aumenta la versatilidad y la adaptabilidad, dado que los desarrolladores pueden hacer los ajustes necesarios de forma inmediata durante el proceso de desarrollo.
- Las iteraciones rápidas reducen el periodo de desarrollo y agilizan la entrega.
- Se fomenta la reutilización del código, por lo que se reduce la programación manual y, en consecuencia, disminuyen tanto la posibilidad de cometer errores como los periodos de prueba.
- Se incrementa la satisfacción del cliente gracias al alto nivel de colaboración y de coordinación entre las partes implicadas, como los desarrolladores, los clientes y los usuarios finales.
- Existe una mejor gestión de riesgos, dado que las personas implicadas cuentan con la capacidad de debatir y abordar las diferentes vulnerabilidades sin necesidad de que se detengan los procesos de desarrollo.
- Se reduce el factor sorpresa, ya que, a diferencia de la metodología de cascada, en el desarrollo rápido de aplicaciones existen integraciones en las fases más tempranas de los procesos de desarrollo de software.

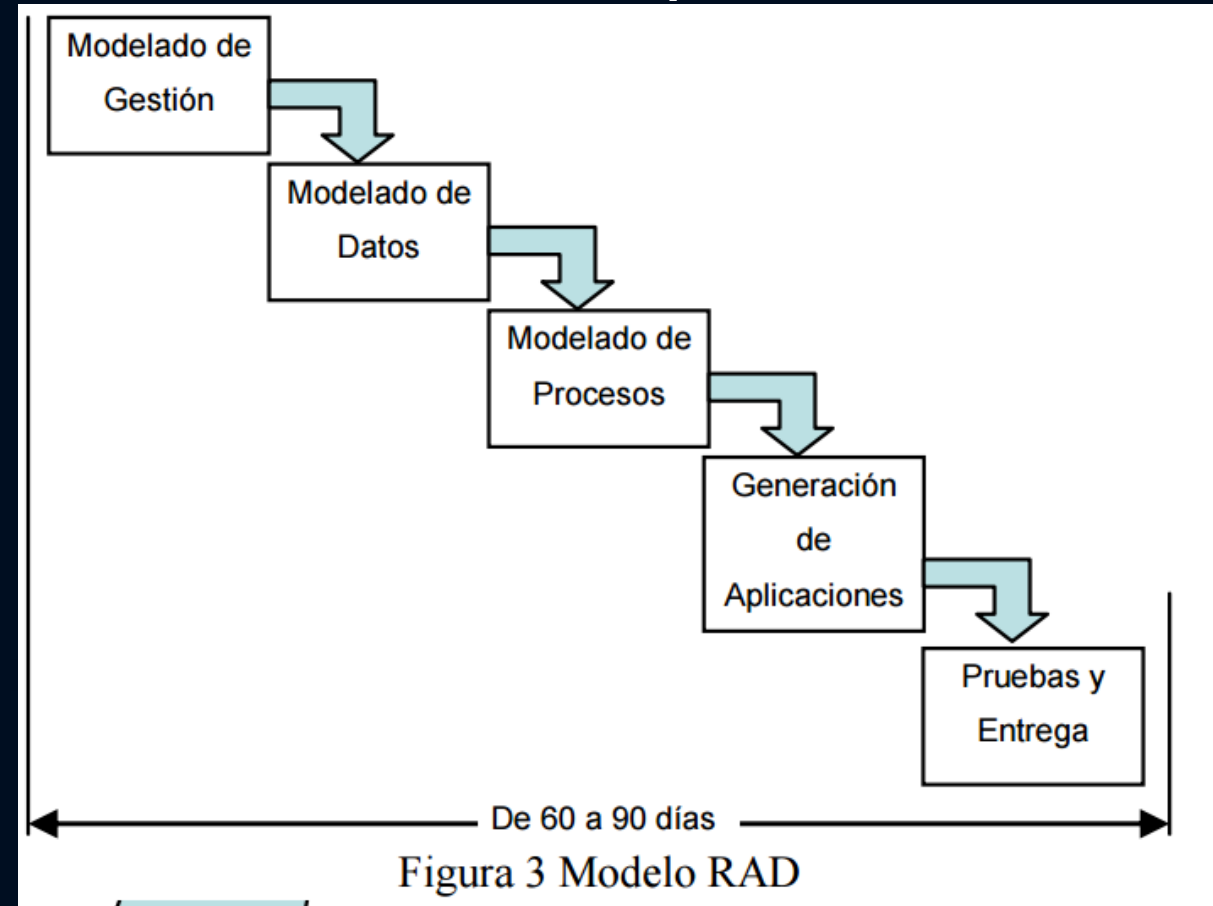
Modelo RAD (Rapid Application Development)

- FASES:
- **1. Definición de Requisitos**
- Durante esta fase, las partes implicadas deben trabajar juntas para definir y concretar los requisitos del proyecto, como los objetivos, las expectativas, los plazos y el presupuesto.



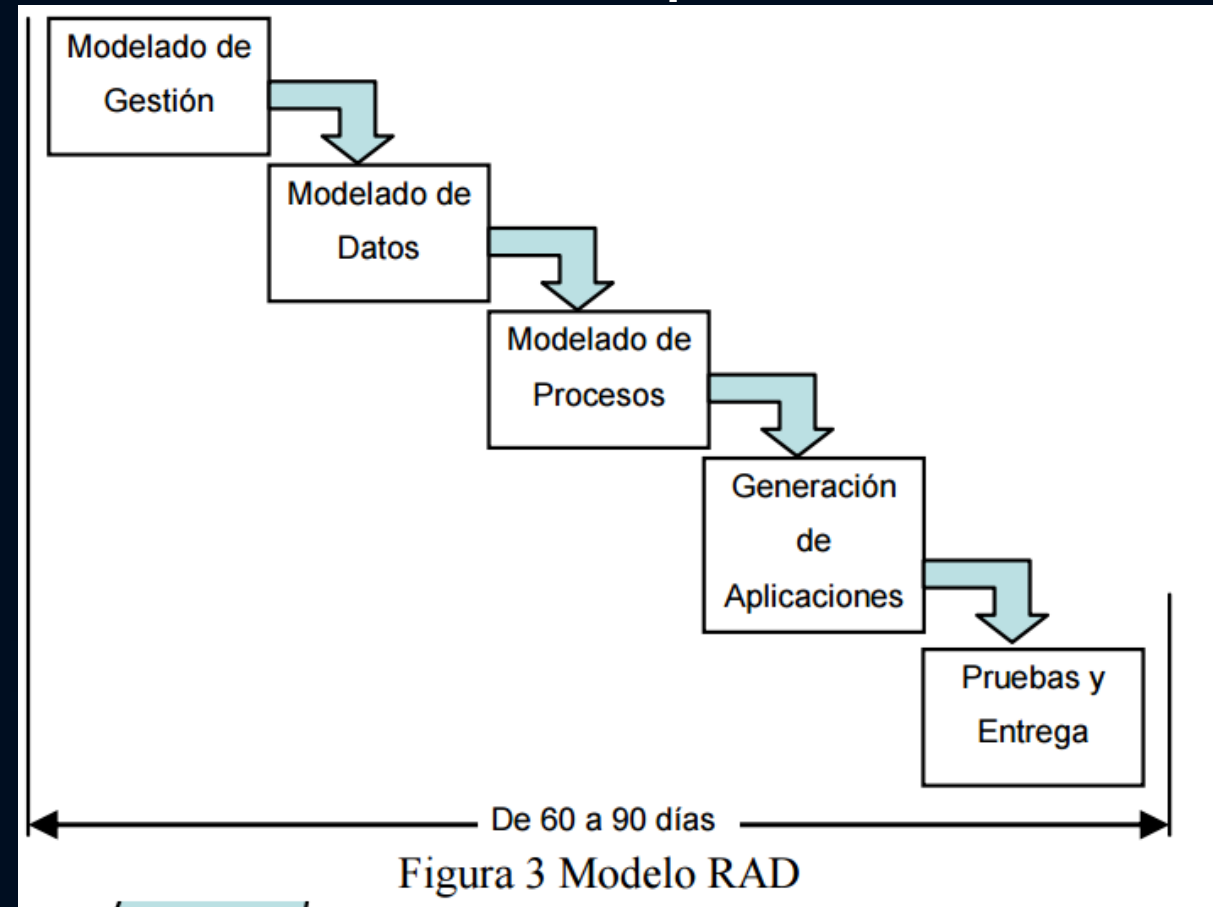
Modelo RAD (Rapid Application Development)

- FASES:
- **2. Diseño de Prototipos**
- En cuanto se haya terminado de definir el alcance del proyecto se puede comenzar la fase de desarrollo. Los diseñadores y desarrolladores trabajarán estrechamente con los clientes con el objetivo de crear y mejorar los prototipos que ya están en marcha hasta que el producto final esté preparado.



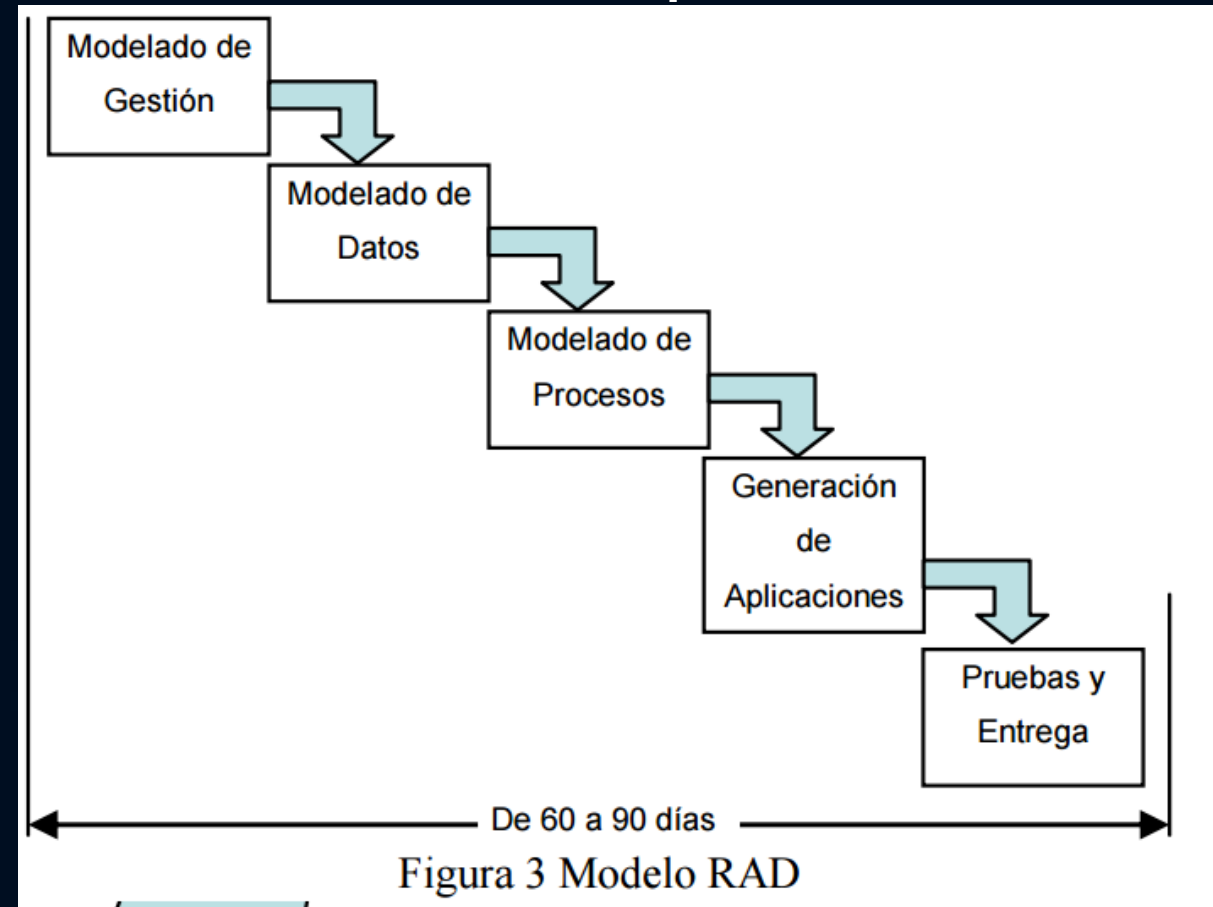
Modelo RAD (Rapid Application Development)

- FASES:
- **3. Recopilar opiniones del usuario**
- En esta fase, los prototipos y los sistemas beta se convierten en modelos de trabajo. Entonces, los desarrolladores recogen la información que proporcionan los usuarios para corregir y mejorar los prototipos y crear un producto con la mayor calidad posible.



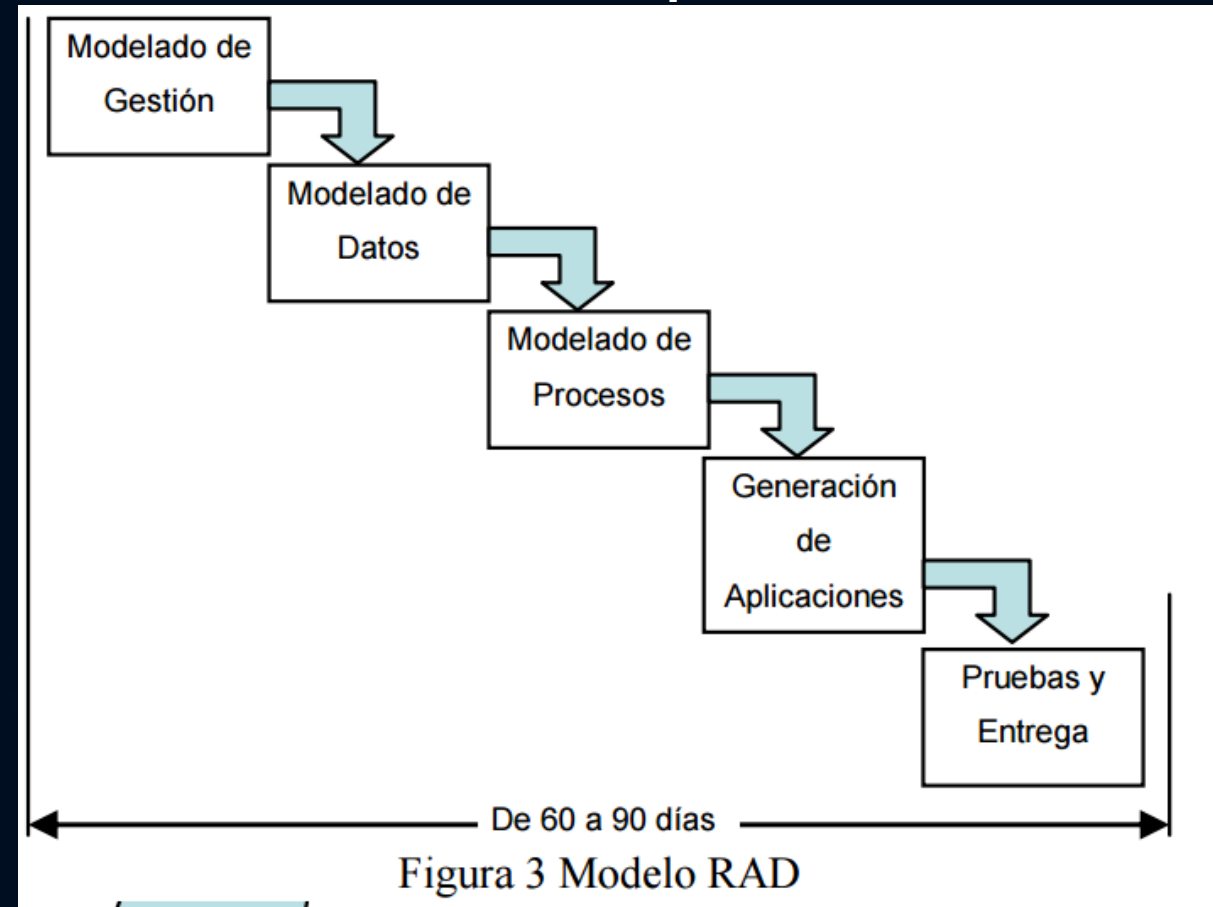
Modelo RAD (Rapid Application Development)

- FASES:
- **4. Pruebas**
- En esta fase es necesario poner a prueba el producto de software y asegurarse de que todos los engranajes funcionan juntos sin problemas para satisfacer las expectativas del cliente. Se sigue incorporando la opinión del cliente a medida que el código se pone a prueba una y otra vez hasta que su funcionamiento sea el correcto.



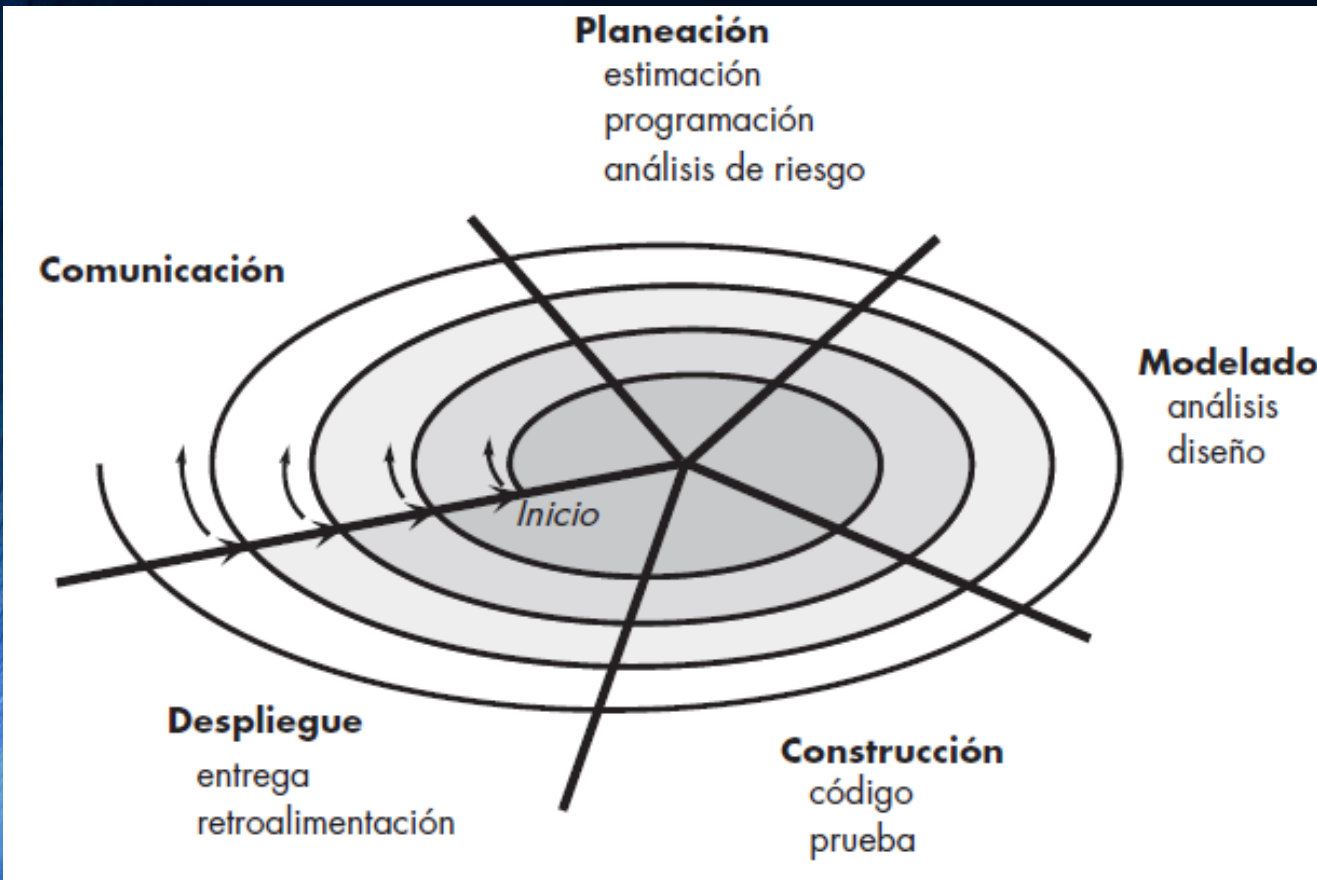
Modelo RAD (Rapid Application Development)

- FASES:
- **5. Presentación y Entrega**
- Esta es la fase donde se presenta y se lanza el producto terminado. En ella se integran la conversión de datos y la formación del usuario.



Modelos de Proceso Evolutivo

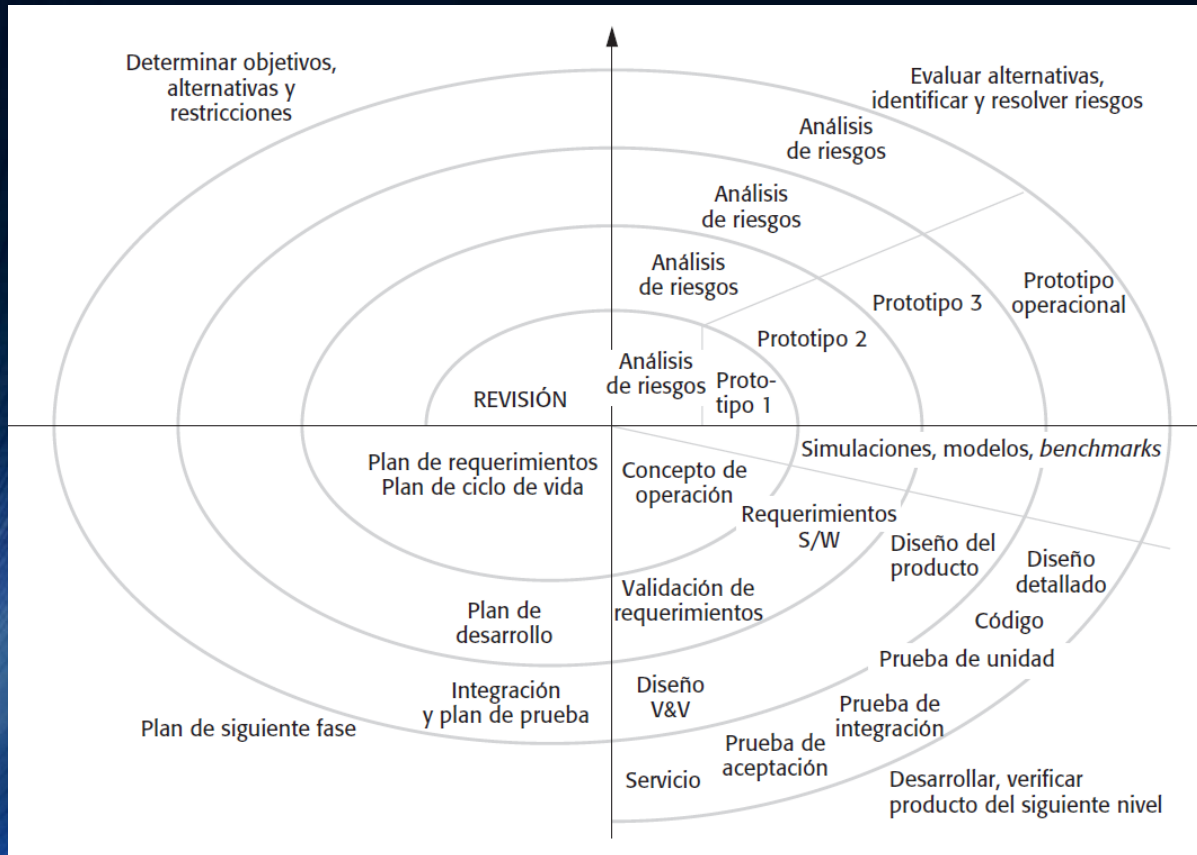
- Modelo de Espiral



- Propuesto en primer lugar por Barry Boehm, el *modelo espiral* es un modelo evolutivo del proceso del software y se acopla con la naturaleza iterativa de hacer prototipos con los aspectos controlados y sistémicos del modelo de cascada.
- Tiene el potencial para hacer un desarrollo rápido de versiones cada vez más completas.
- Con el empleo del modelo espiral, el software se desarrolla en una serie de entregas evolutivas.
- Durante las primeras iteraciones, lo que se entrega puede ser un modelo o prototipo.
- En las iteraciones posteriores se producen versiones cada vez más completas del sistema cuya ingeniería se está haciendo.

Modelos de Proceso Evolutivo

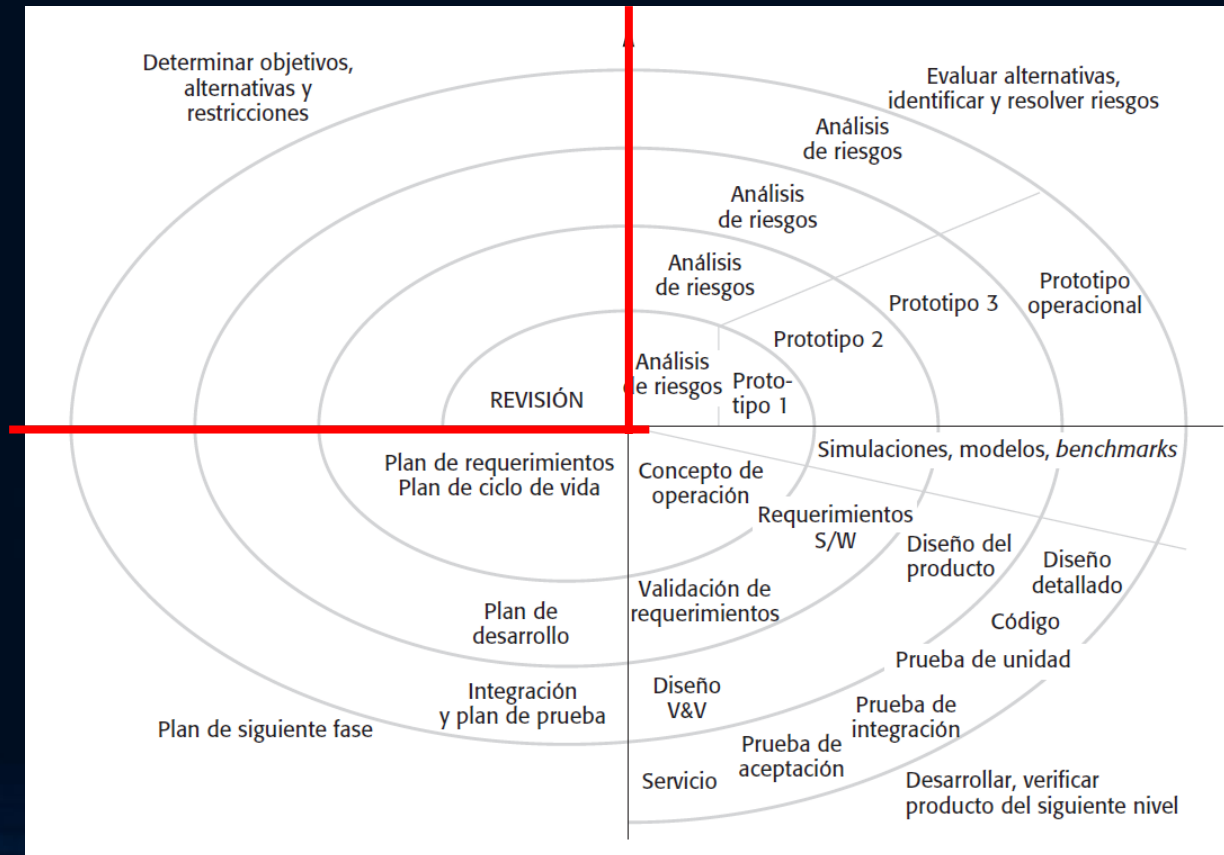
- Modelo de Espiral



- El modelo Espiral es un proceso de software dirigido por el riesgo.
- Cada ciclo en la espiral representa una fase del proceso de software.
- El ciclo más interno puede relacionarse con la factibilidad del sistema, el siguiente ciclo con la definición de requerimientos, el ciclo que sigue con el diseño del sistema, etcétera.
- El modelo en espiral combina el evitar el cambio con la tolerancia al cambio.
- Lo anterior supone que los cambios son resultado de riesgos del proyecto e incluye actividades de gestión de riesgos explícitas para reducir tales riesgos.

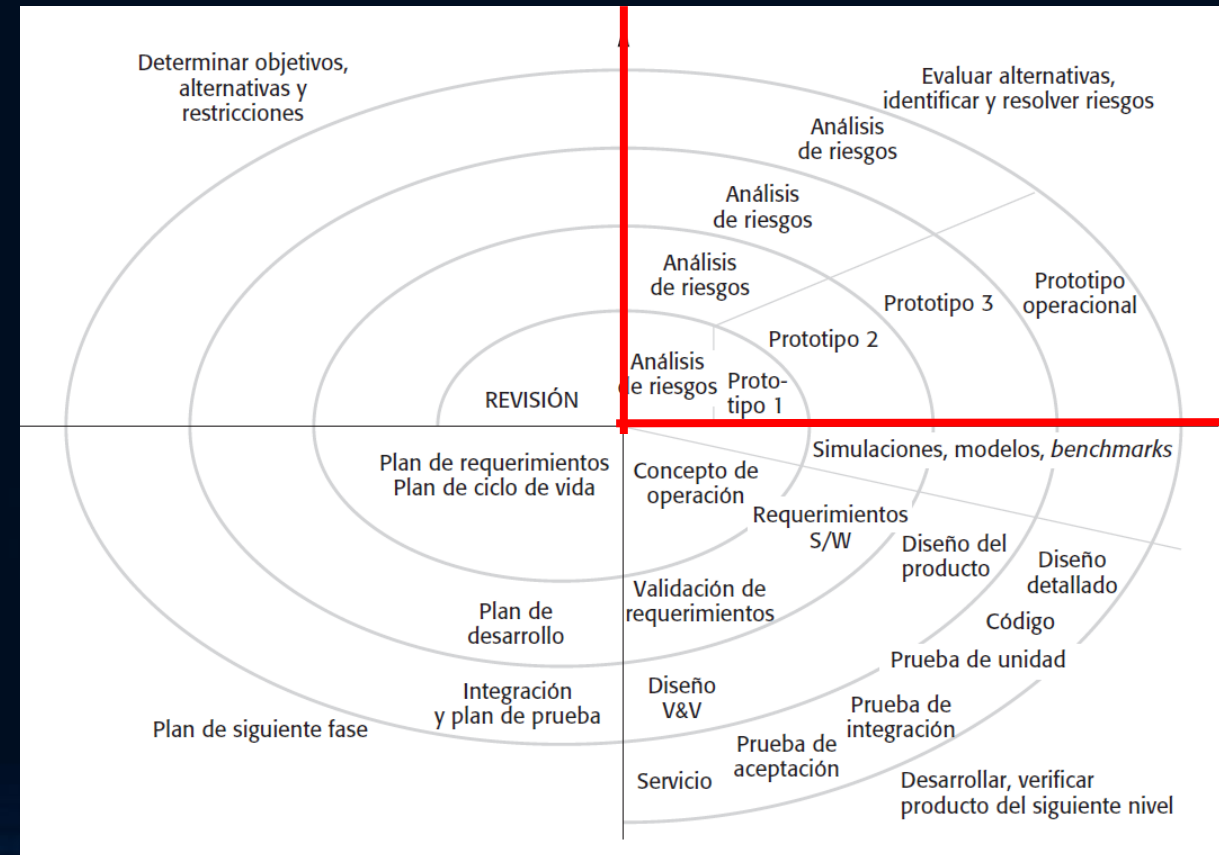
Modelos de Proceso Evolutivo

- Modelo de Espiral
- Cada ciclo se divide en 4 sectores:
- **1. Establecimiento de objetivos.** Se definen objetivos específicos para dicha fase del proyecto. Se identifican restricciones en el proceso y el producto, y se traza un plan de gestión detallado. Se identifican los riesgos del proyecto. Pueden planearse estrategias alternativas, según sean los riesgos.



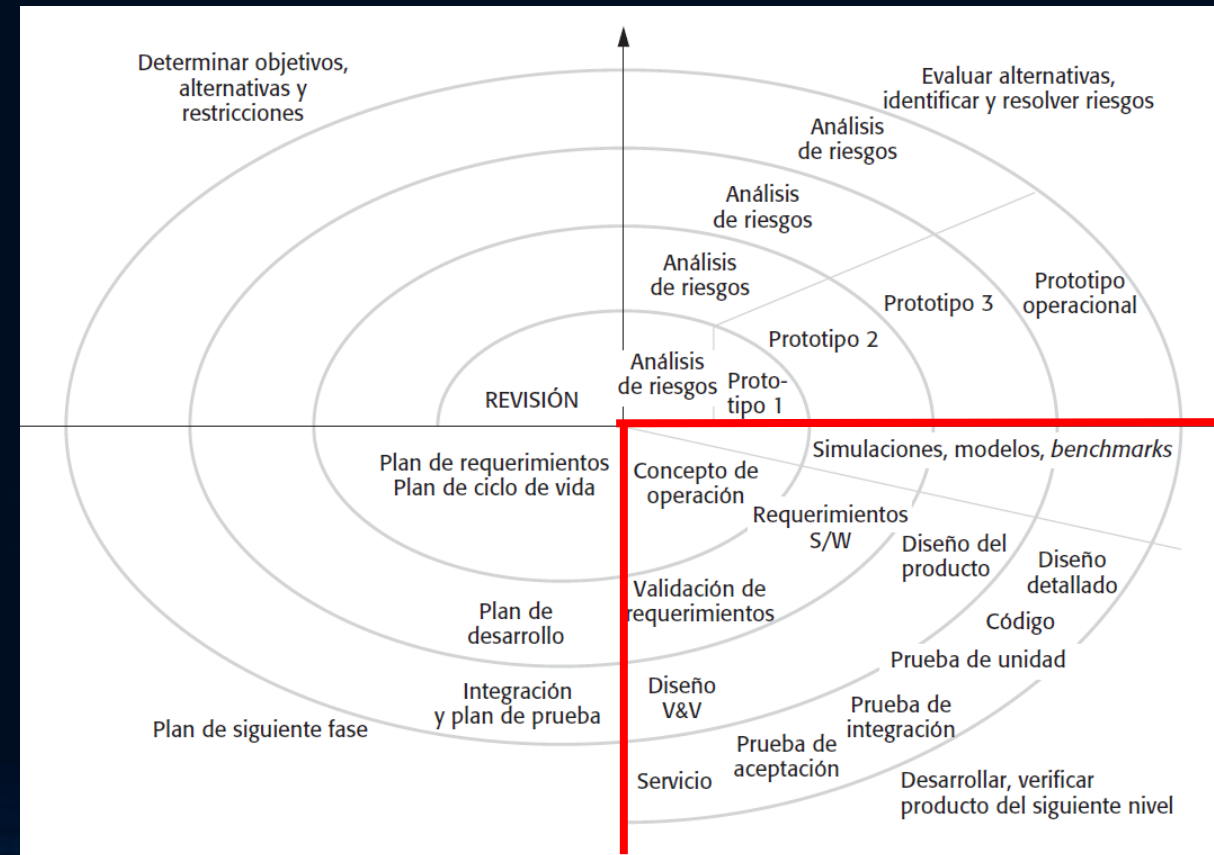
Modelos de Proceso Evolutivo

- Modelo de Espiral
- Cada ciclo se divide en 4 sectores:
- **2. Valoración y reducción del riesgo.** En cada uno de los riesgos identificados del proyecto, se realiza un análisis minucioso. Se dan acciones para reducir el riesgo. Por ejemplo, si existe un riesgo de que los requerimientos sean inadecuados, puede desarrollarse un sistema prototipo.



Modelos de Proceso Evolutivo

- Modelo de Espiral
- Cada ciclo se divide en 4 sectores:
- **3. Desarrollo y validación.** Después de una evaluación del riesgo, se elige un modelo de desarrollo para el sistema. Por ejemplo, la creación de prototipos desechables sería el mejor enfoque de desarrollo, si predominan los riesgos en la interfaz del usuario. Si la principal consideración son los riesgos de seguridad, el desarrollo con base en transformaciones formales sería el proceso más adecuado, entre otros. Si el principal riesgo identificado es la integración de subsistemas, el modelo en cascada sería el mejor modelo de desarrollo a utilizar.



Modelos de Proceso Evolutivo

- Modelo de Espiral
- Cada ciclo se divide en 4 sectores:
- **4. Planeación.** El proyecto se revisa y se toma una decisión sobre si hay que continuar con otro ciclo de la espiral. Si se opta por continuar, se trazan los planes para la siguiente fase del proyecto.

