

Proceso de Desarrollo

- ▶ La estrategia usada para llevar a cabo las actividades de desarrollo hasta llegar al producto
- ▶ Cada actividad tiene sus insumos y entregables, puede requerir herramientas
- ▶ Una actividad puede aparecer más de una vez en el proceso
- ▶ Existen diversos *modelos de proceso* que pueden ser usados

Proceso de Construcción de un Auto



Desarrollo de Software involucra muchas actividades distintas

- Estudio de factibilidad (puede ser muy caro, muy riesgoso o innecesario)
- Ingeniería de requisitos (especificación de requisitos)
- Diseño de Interfaz de Usuario
- Diseño de la Arquitectura
- Diseño detallado (modules)
- Programación (coding)
- Integración (build)
- Verificación (unit testing, system testing)
- Validación (acceptance testing)
- Poner en producción
- Mantención
- Documentación
- Gestión del Proyecto

¿ Se puede desarrollar sin un modelo de proceso ?

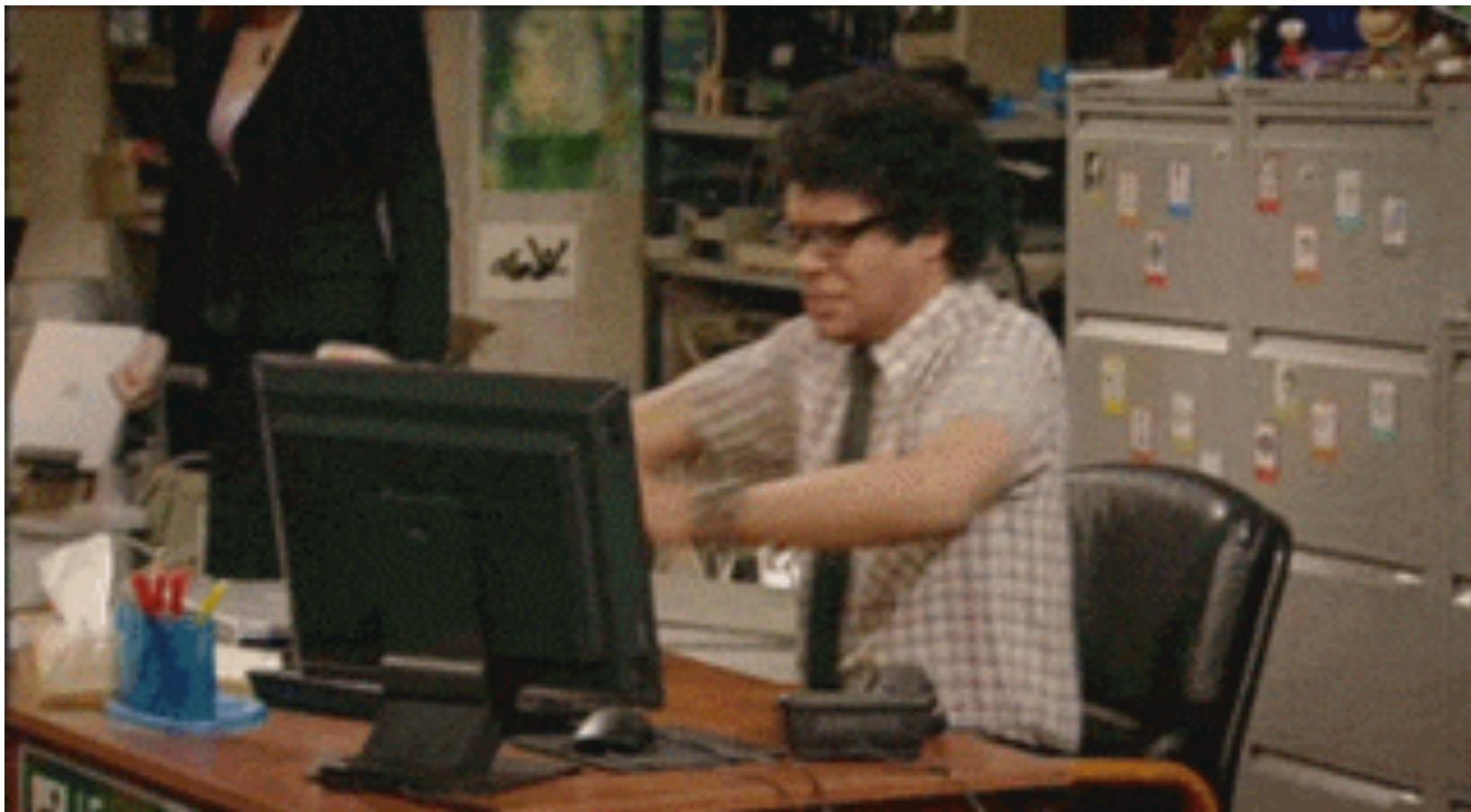
- ▶ Sí, se llama *hacking*
- ▶ programador usa habilidades extraordinarias pero ningún método visible para lograr la meta
- ▶ dependencia muy alta de habilidades personales
- ▶ riesgo muy alto hasta el final



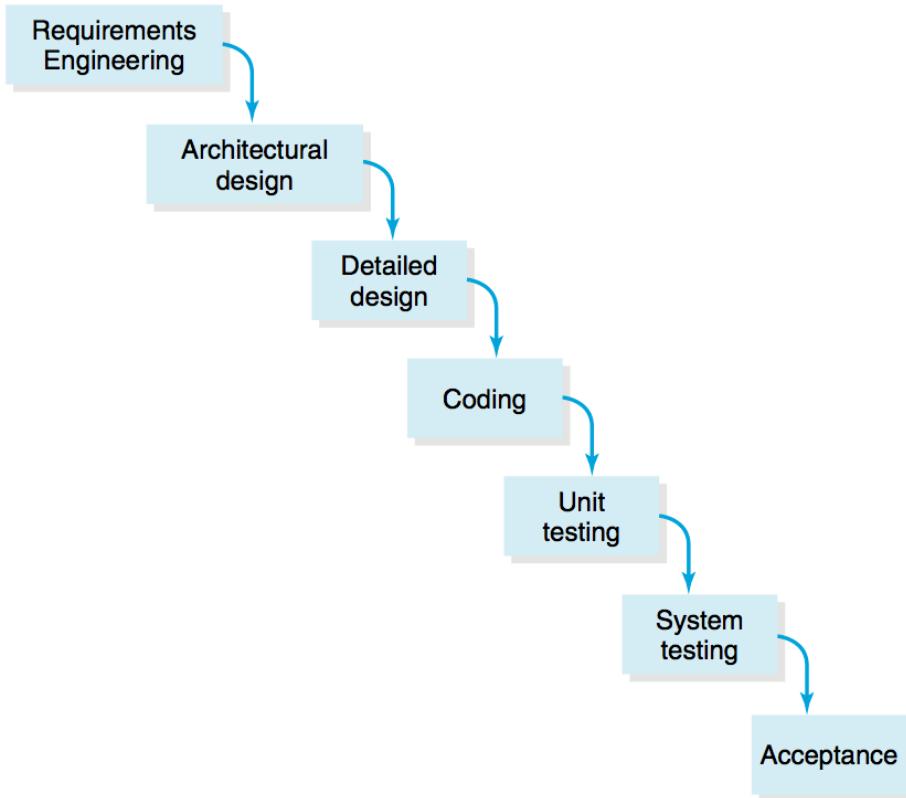
Finalmente ...



Pero mas frecuentemente ...



El Modelo de Cascada (Waterfall)



Stage	Input	Output
requirements engineering	none	requirements specification
architectural design	requirements specification	architectural design
detailed design	architectural design	module specifications
coding	module specifications	coding
unit testing	coding	tested modules
system testing	tested modules	tested system
acceptance	tested system	satisfied client

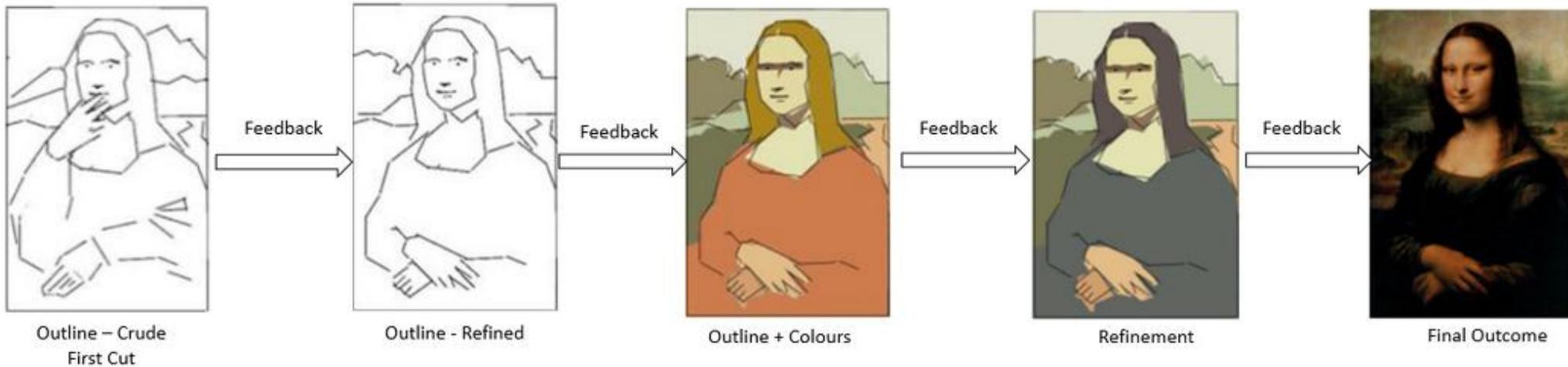
¿ Por qué es popular ?

- ▶ Conceptualmente simple y fácil de entender
- ▶ Similar a procesos de manufactura o construcción
- ▶ Cada fase tiene un input y un output definido
- ▶ Se facilita asignar recursos
- ▶ Se puede saber cuanto llevamos y cuanto falta

Tiene un solo problemita ...

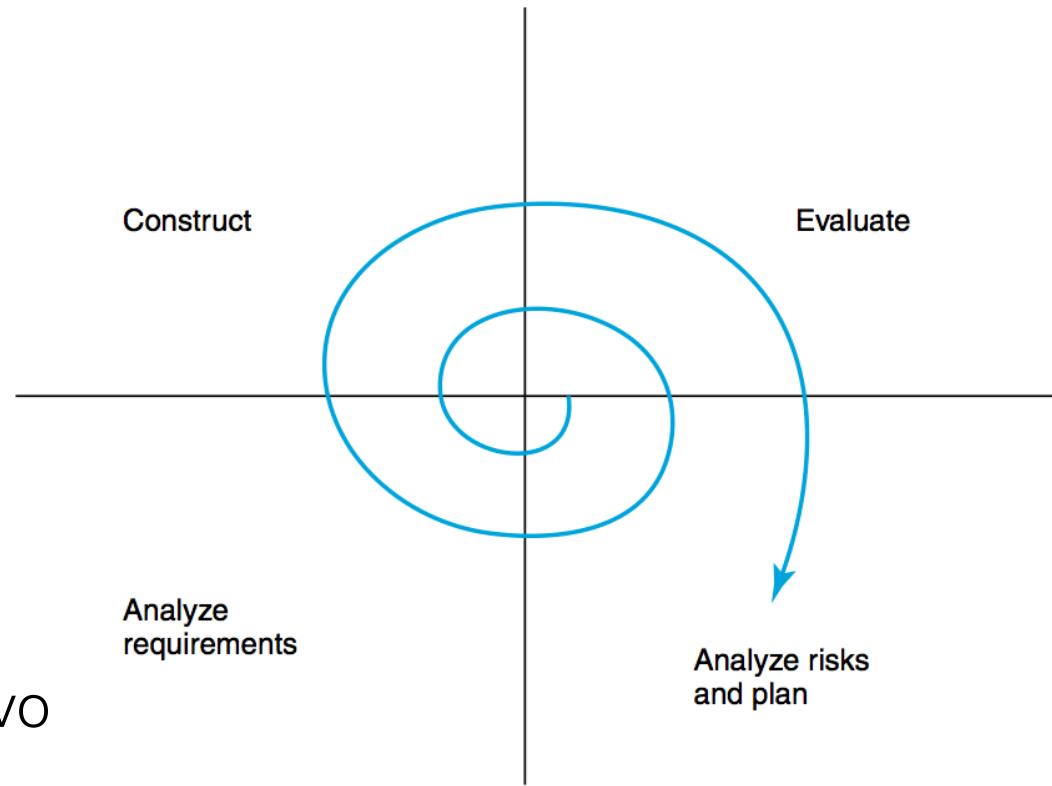
- ▶ No funciona en la práctica !! Salvo proyectos muy cortos
- ▶ no se conocen todos los requisitos al comienzo
- ▶ se conocen pero cambian durante el proyecto
- ▶ Riesgo alto hasta etapas muy avanzadas del proyecto

Procesos Iterativos



- ▶ Actividades de desarrollo se llevan a cabo en una serie de iteraciones
- ▶ No necesariamente cada iteración produce código
- ▶ Ejemplos de procesos iterativos
 - ▶ modelo espiral
 - ▶ modelo unificado (RUP)

Modelo Espiral



- ▶ Enfoque Iterativo
- ▶ Evaluación periódica del riesgo
- ▶ No hay número definido de iteraciones
- ▶ Construcción puede ser diseño, programación, verificación o validación

Los proyectos de TI suelen ser de alto riesgo



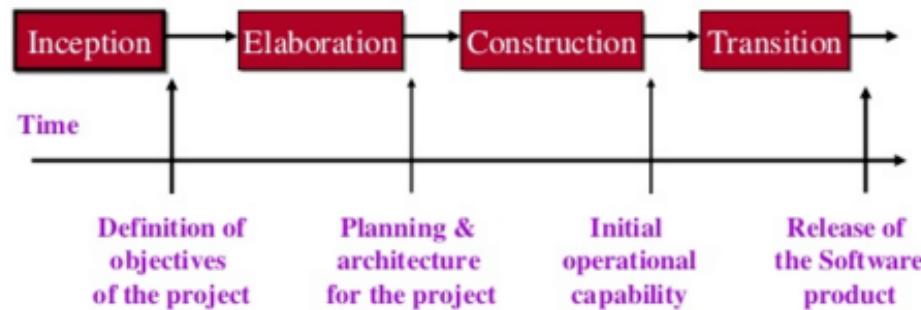
- cliente cambia requerimientos
- no se consideraron algunos requerimientos
- salida de gente del equipo de desarrollo
- una o más de las tareas fuera de plazos
- plataforma de software es muy lenta
- software requiere demasiada memoria
- aparecen nuevas tecnologías de desarrollo de software
- se malentendió un requerimiento
- cambio en el hardware planificado
- un bug que no puede ser reparado
- competidor lanza un producto alternativo

El Proceso Unificado



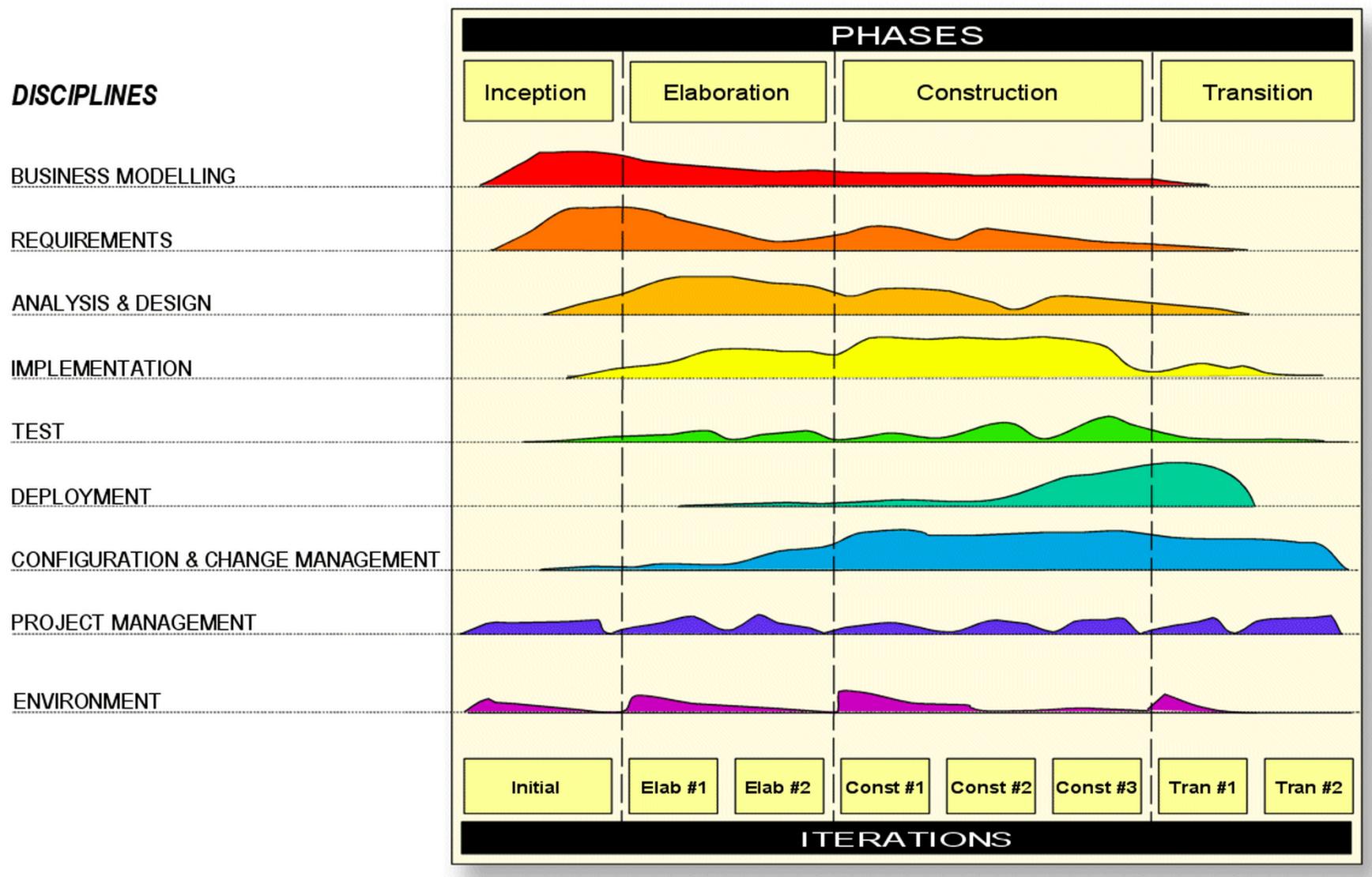
- ▶ Anteriormente conocido como RUP
- ▶ Desarrollo en 4 fases: Incepción, elaboración, construcción y transición
- ▶ En cada fase pueden haber varias iteraciones
- ▶ Se hacen distintas tareas en paralelo

Las 4 fases

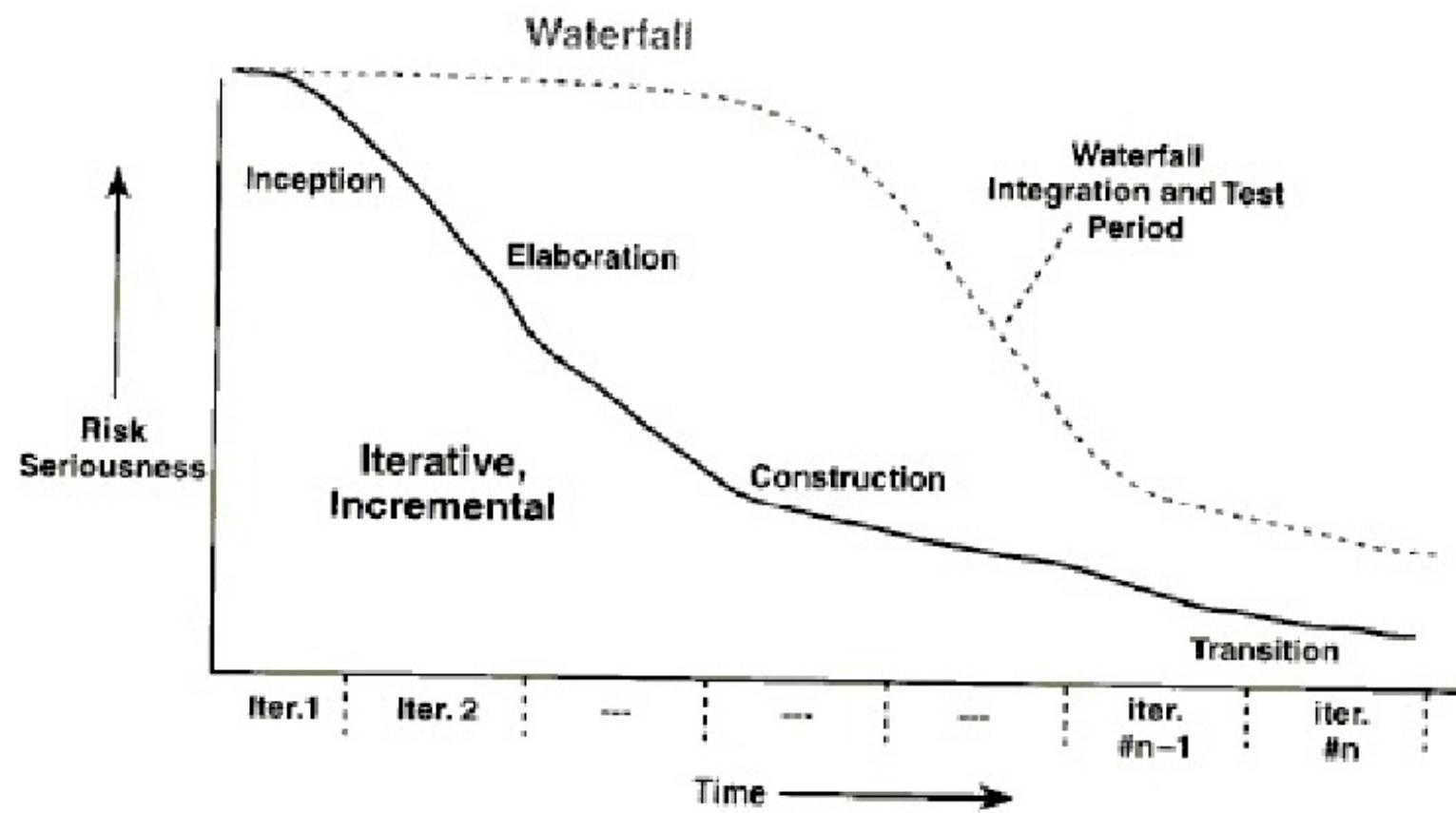


- ▶ Inception - Establecer el caso y la viabilidad del sistema propuesto
- ▶ Elaboration - Establecer la habilidad de construir el sistema dadas las restricciones existentes
- ▶ Construction - Construir un sistema que opere exitosamente (beta)
- ▶ Transition - Entregar un sistema totalmente funcional a los usuarios

Fases, Iteraciones y Tareas



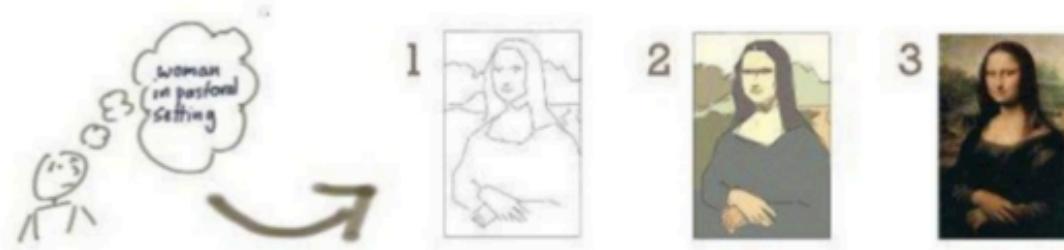
Reducción temprana del riesgo



Procesos Incrementales

- ▶ Software se produce en forma de pequeños incrementos
- ▶ Cada incremento incluye funcionalidades completas (probadas y validadas)
- ▶ Producto se completa con entrega de último incremento
- ▶ Incrementos comunmente asociados a casos de uso (o historias de uso)
- ▶ Es posible tener un proceso iterativo pero no incrementales
- ▶ Los procesos incrementales son iterativos (por naturaleza)

Iterative



Incremental



Iterative & Incremental



Procesos Ágiles

- ▶ Son de carácter iterativo e incremental
- ▶ El Manifiesto Ágil (2001) - los valores centrales
 - ▶ individuos e interacciones por sobre procesos y herramientas
 - ▶ software que corre por sobre documentación
 - ▶ colaboración con el cliente por sobre negociación de contratos
 - ▶ responder a los cambios en lugar de apegarse a un plan

Manifesto for Agile Software Development

A photograph of a diverse group of approximately ten people of various ages and ethnicities, all wearing headsets with microphones, suggesting they are in a video conference or a recording session. They are seated around a large, round, light-colored wooden table in a room with warm lighting and wooden walls.

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Extreme Programming (XP)

- ▶ cliente siempre como parte de equipo de desarrollo
- ▶ casos de uso muy simples (3 fases)
- ▶ cliente especifica casos de prueba
- ▶ software lo más simple posible
- ▶ programación en pares
- ▶ propiedad compartida del código
- ▶ testing permanente

Programación en Pares

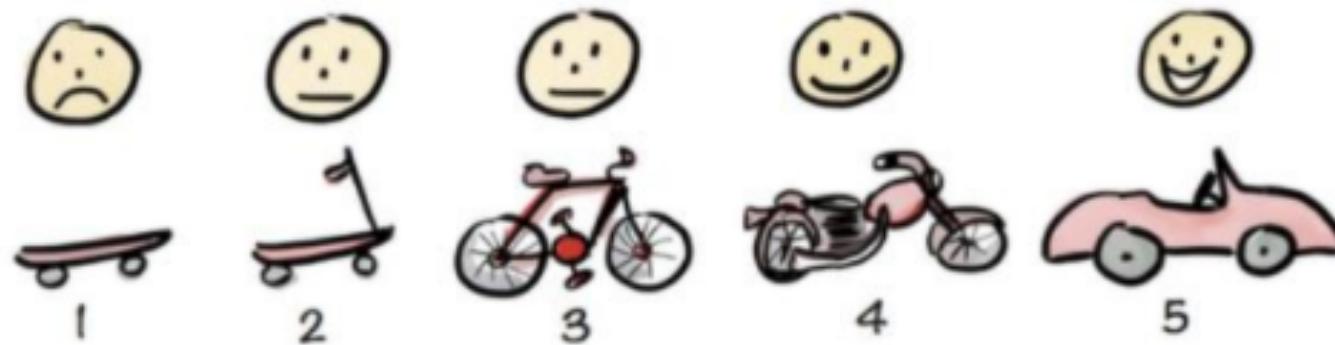
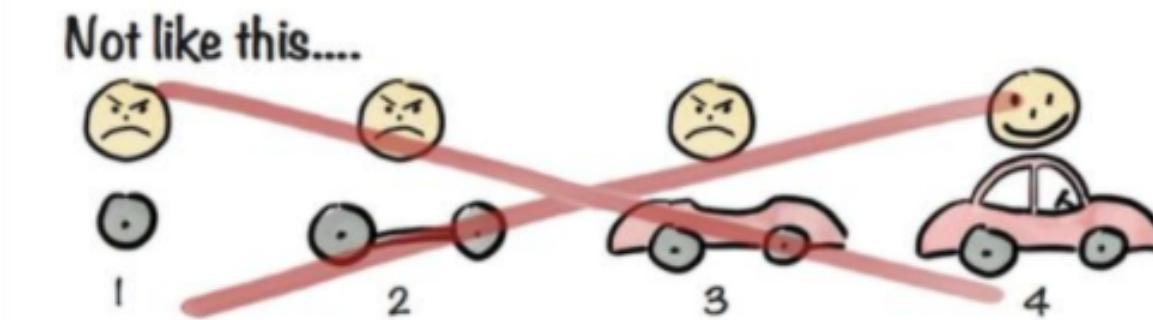


Programación en Pares

https://www.youtube.com/watch?v=oINaUu_wnqo



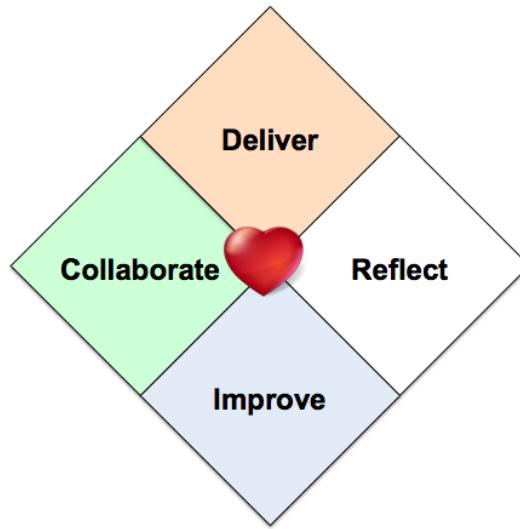
MVP: minimum viable product



Métodos Agiles mas Usados



El Corazón de la Agilidad



- ▶ Idea promovida por Alistair Cockburn, uno de los creadores del Manifiesto Agil <http://heartofagile.com>
- ▶ Han complicado demasiado una idea que era muy simple: colaborar, entregar, reflexionar, mejorar